ORA User's Guide 2011

**Kathleen M. Carley, Jeff Reminga,
Jon Storrick, and Dave Columbus**

June 13, 2011
CMU-ISR-11-107

Institute for Software Research
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Center for the Computational Analysis of Social and Organization Systems
CASOS technical report

*This report/document supersedes CMU-ISR-10-120
"ORA User's Guide 2009", June 2010*

			Form Approved
Report Documentation Page			OMB No. 0704-0188

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE **13 JUN 2011**	2. REPORT TYPE	3. DATES COVERED **00-00-2011 to 00-00-2011**

4. TITLE AND SUBTITLE **ORA User's Guide 2011**		5a. CONTRACT NUMBER
		5b. GRANT NUMBER
		5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)		5d. PROJECT NUMBER
		5e. TASK NUMBER
		5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Carnegie Mellon University,School of Computer Science,Institute for Software Research,Pittsburgh,PA,15213**		8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)		10. SPONSOR/MONITOR'S ACRONYM(S)
		11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT
ORA is a network analysis tool that detects risks or vulnerabilities of an organization?s design structure. The design structure of an organization is the relationship among its personnel, knowledge, resources, and tasks entities. These entities and relationships are represented by the Meta-Matrix. Measures that take as input a Meta-Matrix are used to analyze the structural properties of an organization for potential risk. ORA contains over 100 measures which are categorized by which type of risk they detect. Measures are also organized by input requirements and by output. ORA generates formatted reports viewable on screen or in log files, and reads and writes networks in multiple data formats to be interoperable with existing network analysis packages. In addition, it has tools for graphically visualizing Meta-Matrix data and for optimizing a network?s design structure. ORA uses a Java interface for ease of use, and a C++ computational backend. The current version ORA1.2 software is available on the CASOS website: http://www.casos.ece.cmu.edu/projects/ORA/index.html.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**663**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

Abstract

ORA is a network analysis tool that detects risks or vulnerabilities of an organization's design structure. The design structure of an organization is the relationship among its personnel, knowledge, resources, and tasks entities. These entities and relationships are represented by the Meta-Matrix. Measures that take as input a Meta-Matrix are used to analyze the structural properties of an organization for potential risk. ORA contains over 100 measures which are categorized by which type of risk they detect. Measures are also organized by input requirements and by output. ORA generates formatted reports viewable on screen or in log files, and reads and writes networks in multiple data formats to be interoperable with existing network analysis packages. In addition, it has tools for graphically visualizing Meta-Matrix data and for optimizing a network's design structure. ORA uses a Java interface for ease of use, and a C++ computational backend. The current version ORA1.2 software is available on the CASOS website: http://www.casos.ece.cmu.edu/projects/ORA/index.html.

Table of Contents

viii

xi

ORA | Organizational Risk Analyzer

Latest Version

ORA 2.3.1-f (January 2011)

Contact Information

Carnegie Mellon University
School of Computer Science
Institute for Software Research (ISR)
Center for Computational Analysis of Social and Organizational Systems
(CASOS)
5000 Forbes Avenue
1325 Wean Hall
Pittsburgh, PA, 15213

Questions, Bugs, and Comments

Please contact : Dr. Kathleen M. Carley (kathleen.carley@cmu.edu)

Copyright Notification

Running ORA in Batch Mode

January, 2010
ORA Batch Mode XML
Version 2.0.0

I. Command Line Arguments

The program ora.exe (henceforth called ORA) takes one or more of the following command line arguments as input: (1) Measures File to be used, (2) the Script File to use, and (3) the Log File to use.

If you installed the front-end of ORA, then the executable can be found as:
[ORA installation directory] \include\OraFiles\bin\ora.exe

Likewise, the Measures file can be found as:
[ORA installation directory] \include\OraFiles\xml\ora_xml_measures.xml

Measures File

The Measures File is distributed with ORA and contains an XML encoded description of all the measures available to ORA. This is distributed with ORA and is usually called ora_xml_measures.xml. The following command line parameter specifies the filename:

-measures <filename>

Script File

The Script File is an XML file created by the user and tells ORA where to load the input network data and which reports to generate. Section II - ORA Script XML Specification - describes in detail the format of this XML script. The following command line argument specifies the filename:

-script <filename>

Log File

The Log File is generated by ORA to record the progress and problems of ORA in processing the Script File. The log file is a text file, and the following command line argument specifies its filename:

-log <filename>

Help

A listing of the above parameters is available by using the following command line parameter:

-help

Version

The ORA version can be obtained by the following command line parameter:

-version

II. XML Script

The batch mode script is a user supplied XML document that is input to ORA (see the -script command line parameter above). It tells ORA what network data to load and what reports to generate.

Please note the following general guidelines for the script:

1. camel case capitalization is used for element names: therefore element names begin lower case and the first letter of each word is capitalized. For example, Meta-Matrix is metaMatrix.

2. default values are used: therefore a short, simple script can initially be used to run ORA. Advanced users can change the default values.

The root node is <ora>, which has two child elements <input> and <output>. The input element describes the network data to load, and the output section describes the reports to generate.

Example: an empty script
<ora>
<input/>
</ora>

A. Input Section

The main unit of input to ORA is the Meta-Matrix, and therefore the input section of the XML script tells ORA how to load one or more Meta-Matrices. A Meta-Matrix is simply an organization modeled as a collection of matrices. The entire Meta-Matrix can be loaded from a single data file, or each matrix can be loaded from separate data files with possibly different data formats.

1. Meta-Matrix Specification

3

A Meta-Matrix is specified by the <metaMatrix> element. It accepts a single attribute called id which must be unique across all the Meta-Matrix elements.

The child elements describe the location and format of its matrix data. There are two cases: (1) all data for the Meta-Matrix is stored in a single file, (2) the data for matrices of the Meta-Matrix are located in separate files.

1.1 Meta-Matrix data in a single file

The first case is only possible for a data file in the DyNetML format. In this case, the <filename> and <format> children are required, and the <properties> child is optional. The child elements are described below:

filename : location of the data
format : must by dynetml
properties : [optional] Describes how ORA should interpret the data read from the file, but does not cause ORA to modify the input data in any way. Each graph in the Meta-Matrix is given the property.

Properties are specified with the attributes below, each of which takes values: yes or no

1. sparse: indicates whether the data is sparse

2. binary: indicates whether the data should be treated as binary (i.e. ignore edge weights)

3. symmetric: indicates whether the unimodal graphs should be considered symmetric

4. diagonal: indicates whether the diagonal values of unimodal graphs should be used

Example:
<metaMatrix id="Organization">
<filename>data/organization.xml</filename>
<format>dynetml</format>
<properties binary="yes"/>
</metaMatrix>

1.2 Meta-Matrix data in multiple matrix files

The second case has the Meta-Matrix data located in multiple files. In this case, the script must describe how to assemble the Meta-Matrix from different data files.

Matrix data describes a relationship between two sets of nodes, called the source and target node sets. Therefore, in specifying a matrix, these nodesets must be specified. A unique id for the matrix is also specified. This id must be unique across all matrices in the meta-matrix.

The <matrix> element defines a matrix and has the following case-insensitive attributes:

sourceType : type of the source node set
source : [optional] unique identifier of the source node set
targetType : type of the target node set
target : [optional] unique identifier of the target node set
id : [optional] id of the matrix

The above attributes can be any string.

The sourceType and targetType attributes specify the class or general category of the node sets. By convention, these are taken from Agent, Knowledge, Resource, Task, Location, or Organization, but any string can be used.

The source and target attributes uniquely identify the source and target node sets, and if they are not specified the sourceType and targetType are used. To have more than one node set of a given type (e.g. "blue" agents and "red" agents), the source attribute must be used.

The id uniquely identifies the matrix within the collection of matrices that is the meta-matrix. By default, the id is constructed from the source and target types.

Examples:
<matrix sourceType="agent" targetType="agent" id="reports to"/>
<matrix sourceType="agent" source="employee" targetType="agent" target="employee" id="works with"/>

As in the first case, the <matrix> element must have the child elements <filename> and <format> that describe the location and format of the matrix data. The <properties> element is optional, and pertains only to the matrix.

The <format> element takes one of the following values:

dynetml : DyNetML specified data
ucinet : UCINET binary format (refer to UCINET documentation)
dl : UCINET text format (refer to UCINET DL documentation)
csv : CSV (Comma Separated Values) allows row and/or column labels
raw : space, comma, or semi-colon separated data (no labels allowed)

Example:
```
<metaMatrix id="SampleOrganization">
<matrix sourceType="agent" targetType="agent" id="communication">
<filename>data/organization_agent_agent.dl</filename>
<format>dl</format>
</matrix>
</metaMatrix>
```

Multiple <matrix> elements are used to define the constituent matrices of the Meta-Matrix. Note that <metaMatrix> has no immediate <filename> and <format> children since the data is described under the <matrix> child elements.

Example:
```
<ora>
<input>
<metaMatrix id="SampleOrganization">
<matrix sourceType="agent" targetType="agent" id="social">
<filename>data/organization_agent_agent.dl</filename>
<format>dl</format>
</matrix>
<matrix sourceType="agent" targetType="knowledge">
<filename>data/organization_agent_knowledge.dl</filename>
<format>dl</format>
</matrix>
</metaMatrix>
</input>
</ora>
```

In the example, the Meta-Matrix contains two matrices: the first of type Agent x Agent and named social, and it is populated with data from the specified file of the specified format. The second has type Agent x Knowledge, and no name is specified.

1.3 Multiple Meta-Matrices

Multiple Meta-Matrices can be specified in the <input> section using unique id attributes.

Example: multiple meta-matrices differentiated by id attribute
```
<ora>
<input>
<metaMatrix id="SampleOrganization1">
<matrix sourceType="agent" targetType="agent" id="social">
<filename>data/organization_agent_agent.dl</filename>
<format>dl</format>
</matrix>
<matrix sourceType="agent" targetType="agent" id="friendship">
```

```
<filename>data/organization_agent_agent_friend.csv</filename>
<format>csv</format>
</matrix>
<matrix sourceType="agent" targetType="knowledge">
<filename>data/organization_agent_knowledge</filename>
<format>ucinet</format>
</matrix>
</metaMatrix>
<metaMatrix id="SampleOrganization2">
<filename>data/organization2.xml</filename>
<format>dynetml</format>
</metaMatrix>
</input>
</ora>
```

1.3 Meta-Matrix Directories

If there are multiple DyNetML files in a single directory, then one can
specify the directory and ORA will create a meta-matrix from each file. The
<metaMatrixDirectory> element is used, whose only attribute is name that
specifies the path of the directory. Each file must have .xml extension. A
<transform> element can be used within the element to apply to each
meta-matrix. The id for each meta-matrix is its filename.

Example: load all meta-matrices from a directory
```
<ora>
<input>
<metaMatrixDirectory name="data/sample/">
<transform>
<conform method="union"/>
</transform>
</metaMatrixDirectory>
</input>
</ora>
```

B. Output Section

The second section of the script file is the <output> element, and it
describes which reports to generate. The reports available in ORA and how
to specify them in the ora-script is documented in the file ORA-batch-mode-
parameters.xml, which is an XML specification in Relax-NG.

Ora-Script Report Generation
To tell ORA to create a report, the <reports> child element is used in the
ora-script. The <reports> child element contains a collection of <report>
elements. Each report element specifies any input parameters, the output
filename, and the output file type/format. Each <report> is uniquely
identified by its id attribute.

The <filename> child specifies the name of the output file to create. The filename should NOT have an extension. The <formats> child contains one or more <format> children which specify the format of the report output.

For each report, the possible formats are: text, html, csv, dynetml

Example: generating a report in multiple output formats
```
<report id="intelligence">
<filename>reports/intelligence</filename>
<formats>
<format>text</format>
<format>csv</csv>
</formats>
</report>
```

Any report specific parameters are designated within the <parameters> child. The parameters for each report are documented in the file: ora_xml_measures.xml

Example: specifying report parameters
```
<report id="intelligence">
<parameters>
<numberOfKeyEntities>10</numberOfKeyEntities >
</parameters>
<filename>reports/intelligence</filename>
<formats>
<format>html</format>
<format>csv</format>
</formats>
</report>
```

Example: shorter script if only one output format is specified
```
<report id="intelligence" filename="reports/intelligence" format="text">
<parameters>
<numberOfKeyEntities>10</numberOfKeyEntities >
</parameters>
</report>
```

3. Saving Meta-Matrices

The input Meta-Matrices can themselves be saved to files. This allows for converting between matrix formats and for saving transformed or conformed input data.

Example: saving a specific meta-matrix
```
<ora>
<output>
<metaMatrices>
```

```
<metaMatrix id="sample">
<filename>data/output/sample.xml</filename>
<format>dynetml</format>
</metaMatrix>
</metaMatrices>
</output>
</ora>
```

Only the specified Meta-Matrices will be saved. Individual matrices can be saved using the syntax identical to that of the <matrix> element in the input section. The following, for example, saves the agent x agent matrix:

Example: saving a specific matrix from a meta-matrix

```
<ora>
<output>
<metaMatrices>
<metaMatrix id="sample">
<matrix sourceType="agent" targetType="agent">
<filename>data/output/sample_agent_agent.dl</filename>
<format>dl</format>
</matrix>
</metaMatrix>
</metaMatrices>
</output>
</ora>
```

In addition, network set operations (union, intersection, difference) can be performed on the input networks and then saved. The sample below assumes that there were two meta-matrices specified in the <input> section, namely, A and B, and the set operations all pertain to them. Union and Intersect operate on one or more input meta-matrices, and the parameter method indicates how to handle common edges. The method attribute takes on one of the following values: binary, sum, maximum, or minimum. The difference between two meta-matrices is taken with respect to edges only, and thus no nodes are removed.

Example: saving union of multiple input meta-matrices

```
<output>
<metaMatrices>
<!--Union-->
<metaMatrix id="A Union B">
<filename>data/output/A-union-B.xml</filename>
<format>dynetml</format>
<union method="sum">
<metaMatrix id="A"/>
<metaMatrix id="B"/>
</union>
</metaMatrix>
```

```
<!--Intersect-->
<metaMatrix id="A Intersect B">
<filename>data/output/A-intersect-B.xml</filename>
<format>dynetml</format>
<intersect method="minimum">
<metaMatrix id="A"/>
<metaMatrix id="B"/>
</intersect>
</metaMatrix>
<!--Difference (edges only)-->
<metaMatrix id="A Minus B">
<filename>data/output/A-minus-B.xml</filename>
<format>dynetml</format>
<difference>
<metaMatrix id="A"/>
<metaMatrix id="B"/>
</difference>
</metaMatrix>
</metaMatrices>
</output>
</ora>
```

4. Transform Element

The transform element is used to pre-process a Meta-Matrix or an individual matrix of a Meta-Matrix before generating reports. It transforms its parent element. For example, if the <transform> element is a child of a <matrix> then it applies to that matrix. If it is a child of <metaMatrix>, then it transforms the entire Meta-Matrix: for node set modifications (removeNodes, subsetNodes, conform), these apply to the Meta-Matrix as a whole, but transformations of edges (symmetrizing, dichotomizing) are applied to each matrix independently.

The <transform> element has the following child elements that define the transformation:

<transpose> : transpose the graph

<removeIsolates> : remove isolate nodes

<removePendants> : remove pendant nodes

<symmetrize method=""> : **symmetrize** using one of two methods
"union" : symmetrize using the union/maximum method
"intersect" : symmetrize using the intersect/minimum method
Default is "union".

<subsetNodes> : defines a subset of nodes to create a subgraph
contains one or more of the following child elements:
<nodeSet type=""/> : adds all nodes in the specified node set to the
subset.

The type attribute can have values: agent, knowledge, resource, task,
organization

<node type="" id="" radius=""/> : adds the specified node to the subset
The type attribute is identical to that of the <nodeSet> element.

The id attribute is the node's unique id within the node set.

The radius attribute is optional (by default it is zero), and includes all nodes
within a distance <= radius of the node. This is useful for defining ego node
sets.

<removeNodes> : defines a subset of nodes to remove
contains the same <nodeSet> and <node> child elements as subsetNodes

<conform method=""/> : creates an identical set of nodes across meta-
matrices. This is useful for time-series data to give each Meta-Matrix the
same node sets. The union method adds isolates to achieve identical sets,
and the intersect method removes nodes.

The method attribute can have one of two values:
union: create union of node sets
intersect: create intersection of node sets
Default is "union".

By default the agent, knowledge, resource, task, and organization node
sets are conformed. This can be customized by adding <nodeSet> child
elements to specify specific node sets, as in subsetNodes above.

<dichotomize criteria="" cutoff=""/> : creates a binary graph according to
the criteria and cutoff. Any edges whose value does not satisfy the criteria
and cutoff are given a value of 0, otherwise a value of 1.

The criteria attribute takes one of the following values:
lessThan, lessThanEquals, equals, greaterThanEquals, greaterThan

The cutoff attribute can be any numerical value.

Precedence
The order (i.e. precedence) in which the transformations occur is as
follows: (1) individual matrices, (2) meta-matrices, (3) across meta-
matrices. Each of these usages is in the examples below.

The precedence of operations within a transform (for example, is the graph symmetrized before dichotomizing, etc.) is defined to be the same as they are specified in the <transform> element. Thus, if the <symmetrize> child is before the <dichotomize> child, then the transformation first symmetrizes and then dichotomizes.

Example: remove all agent nodes and the specified knowledge nodes from the meta-matrix
```
<metaMatrix id="test">
<transform>
<removeNodes>
<nodeset id="agent"/>
<node nodesetId="knowledge" id="physics"/>
<node nodesetId="knowledge" id="mathematics "/>
</removeNodes>
</transform>
<filename>data/sample.xml</filename>
<format>dynetml</format>
</metaMatrix>
```

Example: symmetrize a single matrix of the meta-matrix
```
<metaMatrix id="test">
<matrix sourceType="agent" targetType="agent">
<filename>data/aa.csv</filename>
<format>csv</format>
<transform>
<symmetrize method="union"/>
</transform>
</matrix>
<matrix sourceType="agent" targetType="knowledge">
<filename>data/ak.csv</filename>
<format>csv</format>
</matrix>
</metaMatrix>
```

This example creates a sub-Meta-Matrix as follows: a set of nodes is defined, and then the Meta-Matrix induced by the nodes (that is, all the edges incident upon those nodes). The set of nodes consists of all nodes within distance 2 of agent Andy and within distance one of Sam. Because the transform is part of the Meta-Matrix, knowledge, task, or organization nodes can be connected to these agents, and would therefore be part of the set.

Example: define a subset of nodes
```
<metaMatrix id="test">
<transform>
<subsetNodes>
<node nodesetId="agent" id="Andy" radius="2"/>
```

```
<node nodesetId="agent" id="Sam" radius="1"/>
</subsetNodes>
</transform>
<filename>/data/myorg.xml</filename>
<format>dynetml</format>
</metaMatrix>
```

This example makes the agent nodes in the meta-matrices conformable by creating the union of all the agent node sets across the meta-matrices. Currently, only agent nodes can be made conformable.

Example: conform nodesets across meta-matrices
```
<input>
<transform>
<conform method="union"/>
</transform>
<metaMatrix id="org1">
<filename>data/org1.xml</filename>
<format>dynetml</format>
</metaMatrix>
<metaMatrix id="org2">
<filename>data/org2.xml</filename>
<format>dynetml</format>
</metaMatrix>
<metaMatrix id="org3">
<filename>data/org3.xml</filename>
<format>dynetml</format>
</metaMatrix>
</input>
```

Glossary

A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T |
U | V | W | X | Y | Z

A | **Top**

Adjacency Matrix – A Matrix that is a square **actor-by-actor (i=j)** matrix where the presence of pair wise links are recorded as elements. The main diagonal, or **self-tie** of an adjacency matrix is often ignored in network analysis.

Aggregation – Combining statistics from different nodes to higher nodes.

Algorithm – A finite list of well-defined instructions for accomplishing some task that, given an initial state, will terminate in a defined end-state.

Attribute – Indicates the presence, absence, or strength of **a particular connection** between nodes in a Matrix.

B | Top

Betweenness – Degree an individual lies between other individuals in the network; the extent to which an node is directly connected only to those other nodes that are not directly connected to each other; an intermediary; liaisons; bridges. Therefore, it's the number of nodes who an node is connected to indirectly through their direct links.

Betweenness Centrality – High in betweenness but not degree centrality. This node connects disconnected groups, like a Go-between.

Bimodal Network – A network most commonly arising as a mixture of two different **unimodal networks***.

Binarize – Divides your data into two sets; **zero or one**.

Bipartite Graph – Also called a bigraph. It's a set of nodes decomposed into two disjoint sets such that no two nodes within the same set are adjacent.

C | Top

Caesar III – An application for the design of information processing and decision making organizations. An application for the design of decision making organizations at the operational and tactical levels; it takes into consideration cultural differences in coalitions and of adversaries. This tool is under development by George Mason University.

Categorical – An Attribute Type used primarily used for grouping nodes.

Centrality – The nearness of an node to all other nodes in a network. It displays the ability to access information through links connecting other nodes. The closeness is the inverse of the sum of the shortest distances between each node and every other node in the network.

Centralization – Indicates the distribution of connections in the employee communication network as the degree to which communication and/or information flow is centralized around a single agent or small group.

Characteristic Path Length – The median of the average shortest distances (number of links) connecting each employee to all other employees.

Interpretation

Characteristic path length gives a insight into how information spreads. It reflects the communication performance of the entire network and is important for diffusing information and routing information to the right people. A small average path length results in fewer "hops" where individuals or groups don't get information.

Classic SNA density – The number of links divided by the number of possible links not including self-reference. For a square matrix, this **algorithm*** first converts the diagonal to 0, thereby ignoring self-reference **(an node connecting to itself)** and then calculates the density. **When there are N nodes, the denominator is (N*(N-1))**. To consider the self-referential information use general density.

Clique – A sub-structure that is defined as a set of nodes where every node is connected to every other node.

Clique Count – The number of distinct cliques to which each node belongs.

Closeness – node that is closest to all other nodes and has rapid access to all information.

Clustering coefficient – Used to determine whether or not a graph is a small-world network.

Cognitive Demand – Measures the total amount of effort expended by each agent to do its tasks.

Column Degree – see **Out Degree***.

Complexity – Complexity reflects cohesiveness in the organization by comparing existing links to all possible links in all four networks **(employee, task, knowledge and resource)**.

Complementarity – The idea that people seek others with characteristics that are different from and complement their own, aka the idea that opposites attract.

Concor Grouping – Concor recursively splits partitions and the user selects n splits. **(n splits -> 2^n groups)**. At each split it divides the nodes based on maximum correlation in outgoing connections. Helps find groups with similar roles in networks, even if dispersed.

Congruence – The match between a particular organizational design and the organization's ability to carry out a task.

Construct – A reduced form of Construct is found in ORA in the **Near Term Impact Report**.

Continuous – A Attribute Type primarily used for values that are numerical so that statistical analysis can be be performed (Average, Minimum, Maximum, and Standard Deviation).

Count – The total of any part of a Meta-Network **row, column, node, link, isolate, etc.**

CSV – File structure meaning **Comma Separated Value**. Common output structure used in database programs for formatting data.

D | **Top**

Degree – The total number of links to other nodes in the network.

Degree Centrality – node with the most connections. **(i.e. In the know)**. Identifying the sources for intel helps in reducing information flow.

Density –

- **Binary Network:** The proportion of all possible links actually present in the Matrix.

- **Value Network:** The sum of the links divided by the number of possible links. **(i.e. the ratio of the total link strength that is actually present to the total number of possible links)**.

Dyad – Two nodes and the connection between them.

Dyadic Analysis – Statistical analysis where the data is in the form of ordered pairs or dyads. The dyads in such an analysis may or may not be for a network.

Dynamic Network Analysis – Dynamic Network Analysis (DNA) is an emergent scientific field that brings together traditional **Social Network Analysis*** (SNA), **Link Analysis*** (LA) and multi-agent systems (MAS).

DyNetML – DyNetML is an xml based interchange language for relational data including nodes, ties, and the attributes of nodes and ties. DyNetML is a universal data interchange format to enable exchange of rich social network data and improve compatibility of analysis and visualization tools.

E | **Top**

16

Edge – See **Link**.

Node – A *who, what, where, how, why, or thing* that is being studied such as people, agents, organizations, beliefs, expertise, resource, tasks, events, or locations. Node the representation of a single node (a who, what, where, how why item).

Node Class – A set of nodes of one type.

Node Level Metric – is one that is defined for, and gives a value for, each node in a network. If there are x nodes in a network, then the metric is calculated x times, once each for each node. Examples are **Degree Centrality***, **Betweenness***, and **Cognitive Demand***.

Node Set – See Meta-Node.

Eigenvector Centrality – Node most connected to other highly connected nodes. Assists in identifying those who can mobilize others

F | **Top**

FOG – (F)uzzy (O)verlapping (G)roups. Gives a better understanding of individuals spanning groups. Fuzzy groups are a more natural and compelling way of thinking of human social groups.

G | **Top**

General density – The number of links divided by the number of possible links including self-reference. For a square matrix, this **algorithm*** includes self-reference **(an node connecting to itself)** when it calculates the density. When there are N nodes, the denominator is (N*N). To ignore self-referential information use classic **SNA*** density.

Geodesic Distance – A generalization of the notion of a **straight line** to **curved spaces**. In presence of a metric, geodesics are defined to be (locally) the shortest path between points on the space.

Gini coefficient – The measure of inequality of a distribution of income. Uses a ratio with values between 0 and 1: the numerator is the area between the Lorenz curve of the distribution and the uniform (perfect) distribution line; the denominator is the area under the uniform distribution line.

Graph Level Metric – A metric defined for, and gives a value for, the network as a whole. The metric is calculated once for the network. **Examples are Centralization, Graph Hierarchy, and the maximum or average Betweenness.**

GraphML – GraphML is a comprehensive and easy-to-use file format for graphs. It consists of a language core to describe the structural properties of a graph and a flexible extension mechanism to add application-specific data.

Group – a collection of things (entities, nodes, ties, networks).   A group might at times be represented as a meta-node.   Nodes may be classified in to groups on the basis of a shared attribute, type, id-range, label, user selection, etc.   For example, if you have a set of people and know their gender, then their might be two groups - men and women.   In addition the nodes representing those people could be displayed as a meta-node for men and a meta-node for women.   Nodes may be classified in to groups on the basis of a grouping algorithm.   For example, if you have a network showing connections among members of an organization and you run a grouping algorithm it will return clusters of nodes that fit together on some mathematical criteria.   This cluster is a group and can be represented as a meta-node.

H | **Top**

Hamming Distance – Number minimum number of substitutions required to change one string into another string of equal length. (i.e. "**t**on**e**d" and "**r**os**e**s" is 3.)

Heuristics – Problem-solving by trial and error: a method of solving a problem for which no formula exists, based on informal methods or experience, and employing a form of trial and error iteration.

Homophily – (i.e., love of the same) is the tendency of individuals to associate and bond with similar others.

- **Status homophily** means that individuals with similar social status characteristics are more likely to associate with each other than by chance.

- **Value homophily** refers to a tendency to associate with others who think in similar ways, regardless of differences in status.

HTML – **H**yper**T**ext **M**akup **L**anguage.

I | **Top**

In-Degree – The sum of the connections **leading to** an node **from** other nodes. Sometimes referred to **row degree**.

Influence network – A network of hypotheses regarding task performance, event happening and related efforts.

Interstitial – An interstitial (something "in between") is a node situated within but not restricted to or characteristic of a particular group. It provides extra information to the analysis.

Isolate – any node which has no connections to any other node

L | **Top**

Latent Semantic Analysis – Latent semantic analysis (LSA) is a technique in natural language processing, in particular in vectorial semantics, of analyzing relationships between a set of documents and the terms they contain by producing a set of concepts related to the documents and terms.

Lattice Network – A graph in which the links are placed at the integer coordinate points of the n-dimensional Euclidean space and each node connects to nodes which are exactly one unit away from it.

Link – The representation of the tie, connection, relation, link between two nodes.

Link Analysis – A scientific area focused on the study of patterns emerging from dyadic observations. The relationships are typically a form of co-presence between two nodes. Also multiple dyads that may or may not form a network.

Link Class – A set of Links of one type. A set of links of one type can be represented as a meta-link.

Lossy Intersection – Full intersection means that a node/link needs to be in 100% of the networks to be in the resulting network. Lossy intersection at X% percent means that a node/link needs to be in at least X% of the networks to be in the resulting network. Thus, Lossy intersection is a generalization of full intersection.

M | **Top**

Main Diagonal – in a square matrix this is the conjunction of the rows and cells for the same node.

Math Terms

These mathematical terms and symbols are used: Let S be any set:

- $card(S) = |S|$ = the cardinality of S (the cardinality of the node-sets is represented as $|A|$, $|K|$, $|R|$, $|T|$)

- $\Re$ denotes a real number

- Z denotes an integer

Matrix Algebra – The part of algebra that deals with the theory of matrices.

Measure – A measure is a function that maps one or more networks to R^n. Measures are often scalar (n=1) or vector valued with n=|V| or n=|U|.

Meta-Link – The representation of a Group of Links.

MetaMatrix – See Meta-Network

Meta-Network – The representation of a Group of Networks.

Meta-Node – A representation of a Group of nodes.

Monte Carlo – A random optimization of your organization

MRQAP - This describes a report in ORA. See **Reports** for more information. For multivariate cases, if your dependent variable is continous or count data (like in a negative binomial case), you should use MRQAP. If your dependent variable is binary, you should use ERGM (P*) which is forthcoming in ORA. One can perform MRQAP on data that have a dichotomous dependent variable (basically, this is equivalent to using a linear probability model). You do need to interpret your results accordingly.

Multi-Agent System – A loosely coupled network of problem-solver nodes that work together to find answers to problems that are beyond the individual capabilities or knowledge of each node

Multi-node – More than one type of node **(people, events, locations, etc.)**.

Multi-plex – Network where the links are from two or more relation classes.

Multimode Network – Where the nodes are in two or more node classes.

N | **Top**

Neighbors – nodes that share an immediate link to the node selected.

Network – The representation of a set of nodes (including meta-nodes) of one type and the links (including meta-links) of one type between them.

A network N is a triple consisting of two sets of nodes, called U and V, and a set of links E⊂UxV. Thus, we write N=(U,V,E). An element e = (i,j) in E

indicates a relationship or tie between nodes i∈U and j∈V. A network where U=V and therefore E⊂VxV is called **unimodal***; otherwise the network is **bimodal***. We write G=(V,E) for **unimodal*** networks. For our purposes, unimodal networks will not contain self loops, which means that (i,i)∉E for i∈V.

Network Class – A set of Networks on one type. This can also be represented as a Meta-Network.

Network data format – usually comes in the form of NodeSet 1 listed vertically in column A starting in row 2 and NodeSet 2 listed horizontally in row 1 starting with column B. All the intersecting cells are the links between each of the nodes.

Newman Grouping – Finds unusually dense clusters, even in large networks.

Node – A representation of a real-world node (a who, what, where, how, why item.

Node Class – A set of nodes of one type. Note a set of nodes of one type can be represented as a meta-node.

Normalized Sum – Otherwise known as the Average.

Notation:

The following matrix notation is used throughout the document for an arbitrary matrix X:

- $X(i,j)$ = the entry in the ith row and j^{th} column of X

- $X(i,:)$ = i^{th} row vector of X

- $X(:,j)$ = j^{th} column vector of X

- sum(X) = sum of the elements in X (also, X can be a row or column vector of a matrix)

- dich(X) = dichotomize (make binary) X, so that dich(X)(i,j) = 1 iff $X(i,j) > 0$

- X' = the transpose of X

- ~X = for binary X, ~X(i,j) = 1 iff X(i,j) = 0

- X@X = element-wise multiplication of two matrices (e.g. Z=X@Y => Z(i,j) = X(i,j)*Y(i,j))

O | **Top**

ODBC – (O)pen (D)ata (B)ase (C)onnectivity is an access method developed by the SQL Access group in 1992 whose goal was to make it possible to access any data from any application, regardless of which database management system **(DBMS)** is handling the data.

Ontology – **"The Specifics of a Concept"**. The group of nodes, resources, knowledge, and tasks that exist in the same domain and are connected to one another. It's a simplified way of viewing the information.

Organization – A collection of networks.

Out-Degree – The sum of the connections **leading out from** an node **to** other nodes. This is a measure of how influential the node may be. Sometimes referred to as **column degree**.

P | **Top**

Pajek – A visual tool for analyzing large networks (networks having thousands of vertices).

Path – A set of nodes and links that form a single unbroken chain, such that no node or link is repeated. **i.e. A ⇒ B ⇒ C ⇒ D ⇒ E**

Pendant – Any node which is only connected by one link. They appear to **dangle** off the main group.

Percolation Theory – The behavior of connected clusters in a random network.

Pythia – Program used to create Influence Nets. Pythia is a timed influence net application. This tool can downloaded at the following URL: **http://sysarch.gmu.edu/main/software/**.

Q | **Top**

QAP Correlation – Calculates measures of nominal, ordinal, and interval association between the relations in two matrices, and uses quadratic assignment procedures to develop standard errors to test for the significance of association.

QAP is designed as a bivariate test (only two variables). Generally, QAP is perfectly fine for almost any bivariate network problem.

R | **Top**

Random Graph – One tries to prove the existence of graphs with certain properties by assigning random links to various nodes. The existence of a

property on a random graph can be translated to the existence of the property on almost all graphs using the famous **Szemerédi regularity lemma***.

Reciprocity – The percentage of nodes in a graph that are bi-directional.

Redundancy – Number of nodes that access to the same resources, are assigned the same task, or know the same knowledge. Redundancy occurs **only** when more than one agent fits the condition.

Relation – The way in which nodes in one class relate to nodes in another class.

Robustness – Two different definitions:

- **Networks** – Concerned with the reliability (Kim & MÃ©dard, 2004) and continued functioning of a network following an intervention. The robustness of a network is particularly relevant in communication-type and flow-oriented networks. The purpose for understanding robustness of a network has more of a management of the network connotation.

- **Measures** – This meaning has more of a statistical connotation. Studying the robustness of a measure of a network can also be referred to as conducting a sensitivity analysis on the measure. In keeping with the terminology of the most-recently published research in this area, in lieu of using the term sensitivity, we too will use the robustness term, although the terms can be used interchangeably.
A measure is robust if a slight perturbation in its input produces a slight change in its output.

Row Degree – see **In Degree***.

S | **Top**

Scale-Free Network – Some nodes act as **highly connected hubs (high degree)**, although most nodes are of low degree. Scale-free networks' structure and dynamics are independent of the system's size N, the number of nodes the system has. A network that is scale-free will have the same properties no matter what the number of its nodes is.

Self-Loop – An node with a connection to itself.

Shapefile - The GIS Visualizer can view overlay a Meta-Network over a map. The map is a shapefile: a digital vector storage format for storing geometric location and associated attribute information. A "shapefile" commonly refers to a collection of files with ".shp", ".shx", ".dbf", and other

extensions on a common prefix name (e.g., "lakes.*"). For example, Shapefiles could represent water wells, rivers, and lakes, respectively. Each item may also have attributes that describe the items, such as the name or temperature.

Simmelian Ties – Two nodes are **Simmelian Tied** to one another if they are reciprocally and strongly tied to each other and strongly tied to at least one third party in common.

Simulated Annealing – A method of finding optimal values numerically. It's a search method as opposed to a gradient based algorithm. It chooses a new point, and (for optimization) all uphill points are accepted while some downhill points are accepted depending on a probabilistic criteria.

The term **Simulated Annealing** draws its inspiration from metallurgy, where atoms within a metal are heated thereby dislodging them from a metal's internal structure transforming the metal into another atomic state. In this way, your organization is **heated** changing its components in the attempt to arrive at an optimized state.

Slow Measures – As the name implies these measures generally take longer to run.

Small-World Network – Small-World Networks will have sub-networks that are characterized by the presence of connections between almost any two nodes within them.

Social Network Analysis – The term Social Network Analysis (or SNA) is used to refer to the analysis of any network such that all the nodes are of one type **(e.g., all people, or all roles, or all organizations)**, or at most two types **(e.g., people and the groups they belong to)**.

Sphere of Influence – One node's direct relationship with one of it's neighbors as a function of specified path length.

Standard Deviation – The square root of the variance where variance is the average of the squared differences between data points.

.stl file format – This file format is native to the stereolithography CAD software created by 3D Systems. STL files describe only the surface geometry of a three dimensional object without any representation of color, texture or other common CAD model attributes and can us both ASCII and binary representations

Szemerédi's Regularity Lemma – A fundamental structural result in extremal graph theory due to Szemerédi (1978). The regularity lemma

essentially says that every graph can be well-approximated by the union of a constant number of random-like bipartite graphs, called regular pairs.

Table data format – Table data usually contains a column with referencing IDs and headers denoting column names.

Tie – see **Link**

Topology – The study of the arrangement or mapping of the elements (links, nodes, etc.) of a network, especially the physical (real) and logical (virtual) interconnections between nodes.

Total Degree Centrality – The normalized sum of an node's row and column degrees.

Trail – A set of nodes and links that form a single unbroken chain that can revisit nodes but not links. **i.e. A ⇒ B ⇒ C ⇒ A ⇒ D**

Transpose – In linear algebra, the transpose of a matrix A is another matrix A^T (also written A^{tr}, ^tA, or A') created by any one of the following equivalent actions:

- write the rows of A as the columns of A^T

- write the columns of A as the rows of A^T

- reflect A by its main diagonal (which starts from the top left) to obtain A^T

See **The Transpose Wikipedia Entry** for formulas, examples and more information.

Unimodal networks – These are also called square networks because their **adjacency matrix*** is square; the diagonal is zero diagonal because there are no **self-loops***.

Walk – A set of nodes and links that form a single unbroken chain that can revisit both nodes and links. **i.e. A ⇒ B ⇒ C ⇒ B ⇒ D**.

⊞ Getting Started

Welcome to ORA's Help File system! The ORA Help and examples contained herein are written with a specific data set in mind: **Stargate-SG1**. More information on this dataset can be found further down in this section

It is suggested, though not necessary, that the end user work through the examples with this data set. This data set is included with the ORA download and also from the CASOS home page at the following URL: **http://www.casos.cs.cmu.edu/computational_tools/datasets/inter nal/stargate/index2.html**

As an added data set to use, a network model of the **The Tragedy of Julius Caesar** will be employed to demonstrate additional examples. In some cases, both data sets may be used to demonstrate the same technique used on two different data sets for added clarity as well to show ORA's unparalleled versatility in the field of **Dynamic Network Analysis**.

In this data set, we set out to build a network model of the people, resources, tasks, events and knowledge that comprised the world of Julius Caesar as portrayed by The Great Bard, also known as William Shakespeare.

How to use ORA help

If you are not familiar with **Social Network Analysis** we encourage you to read through the beginning sections of the help set. The ORA help set is broken up into six primary parts, which are designed for varying levels of user expertise.

Getting Started : provides an overview of Social Networks and how they pertain to ORA.

The Main Interface : details working with the data within the Networks. This includes loading, saving, merging, performing various math functions on the Networks, generating a variety of networks, and running reports.

The Visualizer : takes these Meta-Networks and turns them into graphical representations. There are tools for creating / deleting nodes and links, displaying information in different formats, and helping to analyze the data.

ORA Measures : explains the various measures used in ORA. They contain a short explanation. It also describes the **type, input, and output**.

ORA Reports : contains a description of what each report is and what it will tell you about your data.

Lessons : are designed to teach specific functions and how to carry out many types of useful network analytical tasks that can be accomplished by using ORA.

What is ORA?

An Overview

The **Organizational Risk Analyzer** (ORA) is a statistical analysis package for analyzing complex systems as **Dynamic Social Networks**.

Many complex systems such as organizations, intra-state alliances, food webs, etc. can be represented as an ecology of interlinked networks. Within ORA any complex system is represented as a Network. What follows is a general description of ORA's primary capabilities. Specific instruction is provided under the correlating folders in this help system.

See Basic Terms for additional definitions relating to Social Networks.

The ORA Visualizer

The ORA Visualizer renders conceptual images of social networks. **Nodes such as Agents, Task, Knowledge and Organizations** represent real-world **Entities** like the president of a company, driving to the airport, or how to wash your clothes. Nodes which share the same attributes can be further grouped together creating Meta-Node. **Links** connect nodes that share a direct relationship. Such relationships are derived from the **Network**. ORA's single unit of data input, and are referred to as graphs. The ORA visualizer is interactive. You can **zoom, rotate, isolate, add and remove Meta-Nodes**, and much more.

Reports

ORA can run many reports: **Risk, Intelligence, and Sphere of Influence** to name a few. Multiple organizations can be compared against each other, network structure can be optimized, subgroups within a network can be identified, and scenarios involving the removal of nodes or links can be examined. Reporting capabilities are constantly being refined and updated.

Charts

Four chart types are available: **Bar Chart, Scatter Plot, Histogram, and Heat Map**. Each one in turn presents a different statistical profile of a selected nodeset. Examples of these reports and how to access them can be found under Basic Usages.

Overview

Contact Information

Kathleen M. Carley

CASOS, Carnegie Mellon University

kathleen.carley@cs.cmu.edu

tel: 412 268 6016

ORA

ORA is an analysis tool designed to help the user evaluate one or more networks. It can be used to assess the nature of, features of, change in, and determinants of complex networks. A large variety of networks can be assessed including, but not limited to, social networks, activity networks, task networks, knowledge networks, supply chains, and communication networks. Using ORA questions such as the following can be addressed: what is critical, are there groups of interest, are there patterns of interest, how might interventions impact the network, who is critical, are there emerging groups, how is the network changing.

What is a network? In any data there are relations among things such as two people are cousins, or a set of people are all members of the same club. These networks might be psychological or cultural as when two people share the same belief. They might be physical as in two resources being in the same location or two computers being connected by a line, or they might be social as in people being related. There are many reasons why networks exist. Networks are ubiquitous. Everyone and everything is constrained and enabled by the networks in which they are embedded. Everyone and everything is embedded in multiple networks; e.g. you are connected to some people due to work, others due to school, others due to your kids, and so on. ORA lets you visualize, assess, and reason about these networks.

What kinds of networks?

ORA can assess any type of network. Anything that can be represented as a set of nodes and relations can be assessed. Typical networks are: social networks (who interacts with whom), financial networks (who lends money to whom); gant charts (what task needs to be done before what), supply chains (what resources are needed to build what other resources)., semantic networks (what ideas are linked to what ideas), proximity networks (what states border on what states), and so on. ORA can assess any data that can be represented as nodes and relations regardless of what the nodes are or what the relations are.

What if there are many networks?

ORA can be used to assess the interlocks among networks. For examples, imagine that the following networks exist: who works with whom; who has what skills, what skills are needed for what tasks, who is doing what tasks then ORA can assess whether in fact the right people have been assigned to those tasks or where there are problems because there is insufficient skills available for the task. ORA can assess "multi-mode multi-link" data.

What if the networks change?

ORA can be used to assess change in networks over time. Over time, the shape or topology of a network might change; e.g., start up companies move from a collaborative to a hierarchical structure, terrorists change what weapons they use to attack sites, scholars cite different papers, and so on. Consequently, over time, who or what is critical in a network might change. ORA can assess change in networks, and forecast possible ways in which the network might change in the future.

What if information is incomplete?

ORA can assess networks even when information is incomplete. ORA has been built to gracefully degrade so that it calculates all and only those metrics for which there is information.

Illustrative applications

ORA has been applied in numerous settings. Examples include the following. Assessment of team-x leadership structure at NASA. Comparison of terror networks. Identification of vulnerabilities in insurgency and terror networks. Assessing change in political elite structures in Korea and early identification of emergent leaders. Designing teams for wargames. Assessing the impact of turnover and retirement on public health organizations in Duchess County, NY. Identifying reports with AIS data. Organizations where ORA has been used include: AFIT, SOCOM, USMA, CIA, many universities.

Tool Evolution

There are still many challenges in the area of dynamic network analysis. Key challenges include handling streaming data, locating confidence intervals for metrics, uncertainty assessment, and so on. As these challenges are met, new algorithms are implemented, tested and added to ORA thus further meeting the user's needs.

Social Network Analysis (SNA)

Social Network Analysis is a scientific area focused on the study of relations, often defined as social networks.

In its basic form, a social network is a network where the people **entities** and the relations **links** are a form of connection such as friendship. Social Network Analysis takes graph theoretic ideas and applies them to the social world.

The term **social network** was first coined in 1954 by J. A. Barnes **(see: Class and Committees in a Norwegian Island Parish)**. Social network analysis is also called network analysis, structural analysis, and the study of human relations. SNA is often referred to as the science of **connecting the dots**.

Today, the term Social Network Analysis **(or SNA)** is used to refer to the analysis of any network such that all the entities are of one type **(e.g., all people, or all roles, or all organizations)**, or at most two types **(e.g., people and the groups they belong to)**. The metrics and tools in this area, since they are based on the mathematics of graph theory, are applicable regardless of the type of nodes in the network or the reason for the connections.

For most researchers, the entities are actors. As such, a network can be a cell of terrorists, employees of global company or simply a group of friends. However, entities are not limited to actors. A series of computers that interact with each other or a group of interconnected libraries can comprise a network also.

Where to find out more on SNA

- *Scott, John*, 2000, Social Networks, Sage (2nd edition)

- *Wasserman, S. & K. Faust*, 1994, Social Network Analysis: Methods and Applications

DyNetML Networks

- **DyNetML**
- **DyNetML** is an xml based interchange language for relational data including nodes, ties, and the attributes of nodes and ties. DyNetML is a universal data interchange format to enable exchange of rich social network data and improve compatibility of analysis and visualization tools
- **Uni-Modal Networks**
- Uni-modal networks are not changed (e.g. an AA does is NOT symmetrized), but bi-modal networks are **symmetrized** in that the links from two or more networks are combined using the maximum method.
- **Bi-Modal Networks**
- Bi-modal networks are combined. e.g., if there is an AK and a KA network, then the KA is transposed to AK2 and then AK and AK2 are unioned with the Maximum link weight from each (a,k) or (k,a) being used.
- Network names are normalized to have IDs agent x agent, agent x knowledge, etc..

ORA and DNA

In general, you may want to use ORA in conjunction with other computational tools to advance DNA theory. The **CMU CASOS tools** that work with ORA to form tool chains are **AutoMap** (extracts networks from texts) and various DNA simulators including both **Construct and DyNet**. These tools are often used in a number of real world applications, including the following areas:

- Designing adaptive teams for Command and Control Networks

- Estimating the impact of organizational downsizing

- Estimating the effectiveness of new structures

- Evaluating risk in organizational designs

- Examine impact of IT effectiveness

- Impact analysis of actions in asymmetric warfare simulation

- Impact analysis of weaponized biological attacks on cities

ORA is interoperable with a number of other SNA and link-analysis tools: **UCINET, KeyPlayer, and Analyst Notebook**. Additional information is listed under data import and export.

Where to learn more about CASOS tools and this tool chain:

Kathleen M. Carley, Jana Diesner, Jeffrey Reminga, Maksim Tsvetovat, 2004, <u>An Integrated Approach to the Collection and Analysis of Network Data</u>, In Proceedings of the NAACSOS 2004 Conference, Pittsburgh, PA

Kathleen M.Carley, 2004, <u>Estimating Vulnerabilities in Large Covert Networks Using Multi-Level Data</u>, In Proceedings of the NAACSOS 2004 Conference, Pittsburgh, PA

Kathleen M. Carley, 2003, <u>Dynamic Network Analysis in Dynamic Social Network Modeling and Analysis: Workshop Summary and Papers</u>, Ronald Breiger, Kathleen Carley, and Philippa Pattison, (Eds.) Committee on Human Factors, National Research Council, National Research Council. Pp. 133-145, Washington, DC.

Kathleen M. Carley, Jana Diesner, Jeffrey Reminga, Maksim Tsvetovat, 2005-forthcoming, <u>Toward an Interoperable Dynamic Network Analysis Toolkit, DSS Special Issue on Cyberinfrastructure for Homeland Security: Advances in Information Sharing, Data Mining, and Collaboration Systems</u>.

CASOS: Center for Computational Analysis of Social and Organizational Systems (http://www.casos.cs.cmu.edu/index.html)

System Requirements

ORA performs best on machines that meet or exceed the following specifications:

Spec	Windows	Mac (running XP)
System	XP	OSX 10.4.x
RAM	256 MB	1GB
Processor	Pentium 4	Intel Core 2 Duo
Speed	1.0 GHz	2.0 GHz
Java	6.0	6.0
Extra		Parallels for Dual Booting

NOTE : **When working with extremely large data sets, increasing processing speed and RAM is highly recommended.**

Files Formats

ORA can read in a multitude of file formats. Below is a list, with examples of the code, of those formats.

DyNetML – **.dl** – **UCINET** – **comma delimited (.csv)** – **text files (.txt)** – **Pajek (.net)** – **GraphML** – **GMU model (.stl)** – **Pathfinder (.lo)**.

DyNetML (the native ORA xml format): DyNetML is an xml based interchange language for relational data including nodes, ties, and the attributes of nodes and ties. DyNetML is a universal data interchange format to enable exchange of rich social network data and improve compatibility of analysis and visualization tools.

```
<node id="col_jack_o'neill">
  <properties>
    <property name="ally" type="string" value="yes"/>
    <property name="gender" type="string" value="male"/>
  </properties>
</node>
<graph sourceType="Agent" source="Agent" targetType="Agent"
target="Agent" id="agent x agent" isDirected="true">
  <edge source="A01" target="A11" type="double" value="1"/>
  <edge source="A02" target="A01" type="double" value="1"/>
</graph>
```

.dl: DL (for "data language") is a very powerful and (fairly) simple language that allows the creation of quite complex and large UCINET data sets with minimal data entry. DL language files are plain ASCII text files that can be created with any editor (be sure to store the results as plain text).

```
dl n=9, format=edgelist1
labels:
    A,B,C,D,E,F,G,H,I
data:
    1    1    1
    1    2    1
    1    6    1
        .
        .
        .
    8    7    1
    9    9    1
```

.##h and ##d (UCINet): UCINET datasets are stored in a special (Pascal) format, but can be created and manipulated using both UCINET's and other software tools (text editors and spreadsheets). Each UCINET dataset consists of two separate files that contain header information (e.g. myfile.##h) and the data lines (e.g. myfile.##d). Because of this somewhat unusual way of storing data, it is best to create data sets with the internal spreadsheet editor or DL language tools, or to import text or spreadsheet files and save the results as UCINET files.

.CSV (comma separated values): The CSV ("Comma Separated Value") file format is often used to exchange data between disparate applications. The file format, as it is used in Microsoft Excel, has become a pseudo standard throughout the industry, even among non-Microsoft platforms. File most commonly created from an Excel spreadsheet using the **Save As...** option

- **Each record is one line... but —** A record separator may consist of a line feed (ASCII/LF=0x0A), or a carriage return and line feed pair (**ASCII/CRLF=0x0D 0x0A)... but —** fields may contain embedded line-breaks (see below) so a record may span more than one line.

- Fields are separated with commas.

- Leading and trailing space-characters adjacent to comma field separators are ignored.

- Fields with embedded commas must be delimited with double-quote characters.

- Fields that contain double quote characters must be surrounded by double-quotes, and the embedded double-quotes must each be represented by a pair of consecutive double quotes.

- A field that contains embedded line-breaks must be surrounded by double-quotes

- Fields with leading or trailing spaces must be delimited with double-quote characters.

- Fields may always be delimited with double quotes.

- The first record in a CSV file may be a header record containing column (field) names

```
,A01,A02,A03,A04
A01,0,0,0,0
A02,1,0,0,0
A03,1,0,0,0
A04,1,0,0,0
```

.txt (text files): Otherwise referred to **ASCII files** which contain no extemporaneous formating.

.raw:

.net (Pajek): The file format accepted by Pajek provides information on the vertices, arcs (directed edges), and undirected edges. A short example showing the file format is given below:

```
*Vertices 3
1 "Doc1" 0.0 0.0 0.0 ic Green bc Brown
2 "Doc2" 0.0 0.0 0.0 ic Green bc Brown
3 "Doc3" 0.0 0.0 0.0 ic Green bc Brown
*Arcs
1 2 3 c Green
2 3 5 c Black
*Edges
1 3 4 c Green
```

.graphml: GraphML is a comprehensive and easy-to-use file format for graphs. It consists of a language core to describe the structural properties of a graph and a flexible extension mechanism to add application-specific data. Unlike many other file formats for graphs, GraphML does not use a custom syntax. Instead, it is based on XML and hence ideally suited as a common denominator for all kinds of services generating, archiving, or processing graphs.

```
<xs:annotation>
  <xs:documentation
    source="http://graphml.graphdrawing.org/"
    xml:lang="en">
    Simple type definitions for the new graph attributes.
  </xs:documentation>
</xs:annotation>
```

.stl (GMU model): This file format is native to the stereolithography CAD software created by 3D Systems. STL files describe only the surface geometry of a three dimensional object without any representation of color, texture or other common CAD model attributes and can us both ASCII and binary representations

```
An ASCII STL file begins with the line:
solid {name}
where name is an optional string. The file continues with any
number of triangles, each represented as follows:
```

```
facet normal n1 n2 n3
  outer loop
    vertex v11 v12 v13
    vertex v21 v22 v23
    vertex v31 v32 v33
  endloop
endfacet
where n1-n3 and v11-v33 are floating point numbers in sign-
mantissa'e'-sign-exponent format and concludes with:
endsolid {name}
```

.lo (pathfinder):

Stargate Summit - Synopsis

Story Background

The data for this scenario is based on the Sci-Fi tv show, Stargate SG-1 and specifically the episodes **Summit / Last Stand**. Following is a synopsis of the episodes if you have an questions regarding the dataset.

The DataSets

The Agents – Colonel Jack O'Neill, Major Samantha Carter, Doctor Daniel Jackson, Teal'c, Jacob Carter/Selmak, Lt Elliot, Ren'al, Aldwin, Janet Frazier, Major Mansfield, General Hammond, Lantash, Narim, Travell, Osiris, Yu

The Short of it

The **Tok'ra** are planning an attack which would kill all of the head **Gou'ald** in one fell swoop by infiltrating a summit and poisoning them all simultaneously. With the help of Stargate Command and the U.S. Military they are able to infiltrate the summit. The plan is going fine until new information is disclosed which changes everything.

The Story

The Tok'ra have information regarding a secret Gou'ald summit of all the System Lords. They devise a plan to destroy the organization once and for all by poisoning their symbiotes, but they have need of one thing. A human (tauri) who speaks fluent Gou'ald.

Ren'al (the head of the Tok'ra council) travels to Earth to meet with Stargate Command in order to recruit Dr. Daniel Jackson. Dr. Jackson is a

human who is fluent in Gou'ald. The System Lords have declared a truce for the summit and each System Lord can bring one human servant.

SG-1 journeys to the Tok'ra homeworld, Revanna. Jacob Carter, previously of the U.S. military but now a Tok'ra, presents the plan to Colonel O'Neill and SG-1. He's established himself as a minor Goa'uld in the service of Yu. He can help get Daniel inside Yu's home base by using the Reole memory-altering chemical to convince Yu that he is his closest human slave, Jarren.

Elsewhere, the Goa'uld Zipacna, a servant of the new and very powerful Goa'uld, welcomes a guest, Osiris. Zipacna offers her a position in service of the Goa'uld, Anubis.

Osiris prepares to depart Zipacna's ship for the summit, as Anubis' official representative. Zipacna, meanwhile, has learned the location of the new Tok'ra homeworld, Revanna, from a captured spy and plans to launch a full-scale invasion.

Jacob and Daniel successfully dispose of Jerran, and, using the reole drug, trick Yu into believing Daniel is his faithful servant. Yu and Daniel depart to the summit.

Six of the galaxy's most powerful System Lords are waiting: Ba'al, Olokun, Morrigan, Bastet, Kali the Destroyer, and Svarog. They are waiting, though, for another guest, the powerful Goa'uld who has been dealing them all heavy defeats in battle, but who refuses to make his identity known.

As the attack begins on Revanna SG-1 and SG-17 try to help the Tok'ra, but little can be done. The Goa'uld have dialed into Revanna's Stargate, thusly preventing their escape. Zipacna continues bombing the surface of Revanna, in an attempt to flush out the resistance. As the tunnels collapse many Tok'ra are killed along with Major Mansfield.

Ren'al encrypts the symbiote poison formula onto a data crystal but is killed by falling debris shortly after. Major Samantha Carter takes the data crystal for safe keeping. A large holding tank, where the symbiote Lantash is being kept alive, is destroyed and Lantash takes refuge in the dying Lieutenant Elliot as a host.

Colonel O'Neill, Teal'c and the Aldwin do a scouting run on the surface of the planet only to find Zipacna has amassed an army of Jaffa troops on the surface. Aldwin is killed on their way back to the tunnels. SG-1 regroups but more tunnel collapses prevent their escape to the ring room. Their path to the surface is blocked.

Back at the summit Osiris, the final guest, arrives. Daniel is caught by surprise. Osiris had taken possession of his friend Sarah Gardner a year ago. Osiris recognizes Daniel, but does not reveal this to the System Lords.

In private Daniel communicates her arrival to Jacob. Her arrival presents a large problem. Osiris confronts Daniel Jackson and demands to know what he's is planning. Daniel manages to prick her with his ring containing the reole memory drug and convinces her that he is but Yu's human servant. The drug takes affect and she leaves, somewhat confused. Col. O'Neill and Teal'c find a set of Tok'ra tunneling crystals which they use to grow new tunnels. Major Carter tries to help Lt. Elliot stay alive.

Osiris tells the System Lords she represents the Anubis, who had been believed to be dead for 1,000 years. Anubis has dealt blows to each and every System Lord. Anubis wants entrance back into the System Lords. The Goa'uld vote to allow Anubis to rejoin them, and for Osiris to represent him.

Daniel reports this to Jacob Carter. They both agree that the plan to poison the System Lords must be aborted. If they wipe out the seven most powerful System Lords now, Anubis will be free to conquer the galaxy.

SG-1 and Lt. Elliot make it to the surface but must still must avoid the Jaffa army. They begin the 25-mile walk to the Tok'ra beacon to send a signal to Daniel and Jacob. At the beacon Maj. Carter sends a signal to Jacob and Daniel.

Jacob tells Daniel to get off the space station. But Daniel decides to attempt to kidnap Osiris (still hoping to free her from the Gou'ald). When she arrives, he locks her in the hold and readies to steal the ship. But Yu intervenes saying that his servant Jarren has betrayed him. He releases Osiris. But believing that Yu plotted to kidnap her she stabs Yu. As she prepares to kill Daniel he reveals who he really is. This cause her to hesitate. That, and the attack by Yu gives him the opportunity to escape.

On their return to Revanna Jacob and Daniel are shot down by the Jaffa army and crash. SG-1 finds them shaken but unhurt. Lt Elliot/Lantash, knowing he's dieing anyway, volunteers to sacrifice himself by taking the symbiote poison and wait to be captured. He'll then release the poison clearing the way for SG-1 to escape through the Stargate.

Colonel O'Neill hates to leave his man behind, but sees that there is no other choice. The team goes into hiding. Elliot places the poison in his pocket, and listens as the footfalls of the approaching Jaffa army loom ever closer.

Network Drives

If you are running on a network drive then the help system needs a drive letter in order to work correctly. There are 2 possible solutions to this:

1. Copy the ORA folder to each local machine.

2. If at all possible Map a Network Drive. Right click on the shared folder and select Map Network drive.

Mapping a Network Drive
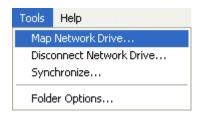
In an Explorer window select **My Network Places**. Find the name of the network which contains ORA.
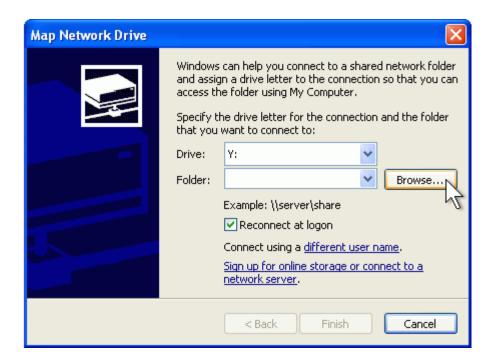
Go back to **My Document**.

From the menu select **Tools ⇒ Map Network Drive**. This brings up the dialog box below.
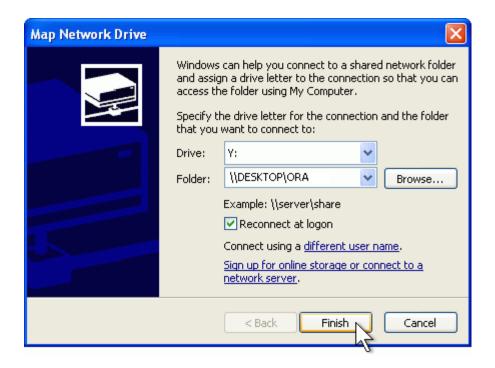
Select the **[Browse]** button in order to select the network drive.

Select the correct network drive and press **[OK]**.
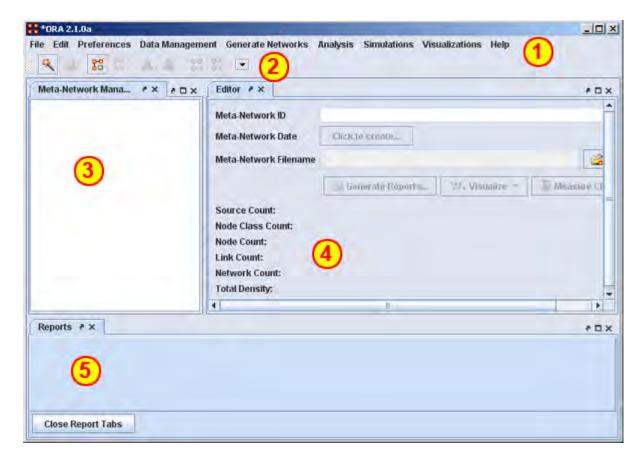
Select **[Finish]**.

Your computer now has the Network drive with ORA mapped on the desktop.

ORA's Main Interface

The ORA interface is organized into **five sections**. The menu, the main icons, and three re-sizable panes which are dockable. On each tab are two buttons: **undock and close**. In the upper right corner of each pane are three buttons: **undock, maximize, and close**. This will allow you to arrange the interface to suit your needs.

1 - The Menu
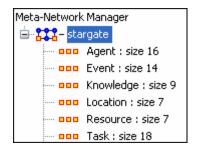
The main source for accomplishing your tasks within the editor. It contains all the functions for managing the Meta-Networks, generating reports and doing analysis.
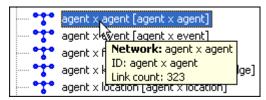
2 - The Icons

Contains the main functions for adding or deleting Meta-Networks, NodeClasses, and Networks and importing existing Meta-Networks. Clicking the down-facing triangle brings up a window which allows you to add buttons to the GUI for the functions you use the most.

3 - Meta-Network Manager Pane

This pane contains a tree directory of the current Meta-Network(s) loaded into ORA. This contains the Meta-Networks currently active plus all their Meta-Nodes and Networks. Clicking on the **plus** icon will expand the Meta-Network showing all the Meta-Nodes and Networks and clicking on the **minus** icon condenses the Meta-Network. Hovering the pointer over a graph it will display information about that Network.

4 - Editor Pane

Contains the basic information about the highlighted Meta-Network such as how many nodes are represented. This panel also allows quick access to ORA's Visualizer, Reports and Chart tools. By selecting the Visualize button, the currently loaded Network will render in the ORA Visualizer, which will load in a separate window.

With the Meta-Network selected panel 2 displays the information regarding the Meta-Network itself. The Meta-Network name and pathway at the top. In the bottom, under statistics, are information regarding the actual Meta-Network including source, node, link, and network counts.

When a Meta-Network is expanded it reveals nodeclasses and networks. Highlighting any of these will reveal information these items.

> *The Info Tab :* Display relevant information whether a nodeclass or network is highlighted.

> *The Editor Tab :* Displays the information in a matrix which can be edited in multiple ways. The data can be directly manipulated in the grid, nodes and attributes can be

With the **Nodes** and **Attributes** buttons you create, delete, or merge nodes and attributes. The **Find** box assists you in locating particular nodes and attributes.

5 - Reports Pane

This pane is initially empty when ORA is first started up. As reports are run they will be contained within this panel.

NOTE : All three panes can be resized to your preference.

Main Interface Tasks

There are a number of tasks that are performed from within the Main Interface to assist in controlling your Meta-Networks, NodeSets, and Networks. These are mostly functions not associated with any of the Menu Items.

Attributes : Adding or Deleting an Attribute and the uses of Attributes.

Attributes Buttons : Controls in the Editor for Attributes.

Clean Button : Helps combine into a single node names with slight variances.

Contextual Menus : Right-Clicking of several items brings up a contextual menu of additional functions for various items.

Creating New Meta-Networks : A brief overview of creating a Meta-Network.

Dating Meta-Neworks : Attaching a date to your Meta-Network

Document/URI Attributes : Using Links as Attributes.

Editor Toolbar : Description of the Editor Toolbar functions..

Filters : Setting Filter parameters for searching nodesets.

Find (Search) : Finding information within the Editor.

Hovering ToolTips : A brief overview of the information given in the tooltips.

ID Names : Description of ID Names and why they are important.

Node Buttons : Description of the buttons in the Editor GUI to work with nodes.

Meta-Network from Excel : Using Excel to create a Meta-Network.

Node Buttons : Tools for working with nodes in the Editor.

Tabs for Editor Node and Network Tabs : Description of the information displayed for a Meta-Network in the Editor and Network tabs.
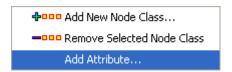
Visualize Buttons : The difference between **Visualize this Network and Visualize Only this Network button.**
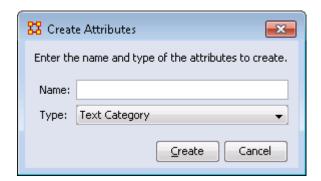
Attributes

An is an extra **container** that can be attached to an Meta-Node. You can add a **value** to this container. To add an attribute to an Meta-Node **right-click** on the Meta-Node name. This will bring up the contextual menu below.

Add Attribute

Hightlight a Meta-Network the right-click to bring up the menu. Then select **Add Attribute...**

Type in the name for your new attribute and select the **Type** form the drop down menu. Your choices are **Text, Text Category, URI, Number Category, Number, or Date**.
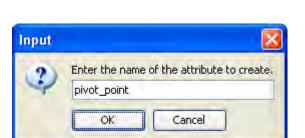
Then click [**OK**]. Let's say we want to add other languages spoken by the nodes.

The attribute you added will be inserted **alphabetically** in the editor.

You can now assign any text to this attribute for any of the rows in the Network which can be used later to help define your Network.</td

45

Add attribute (second option)
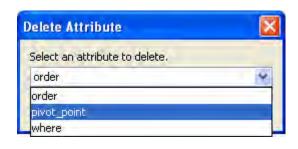
There is a second way to add attributes in the **Editor tab**. Below the matrix is a section called **Attributes** which contains two buttons, **[Create] and [Delete]**.

Selecting **[Create]** brings up a dialog box and asks you to input the name for the new attribute.

The new attribute is then inserted within the matrix.
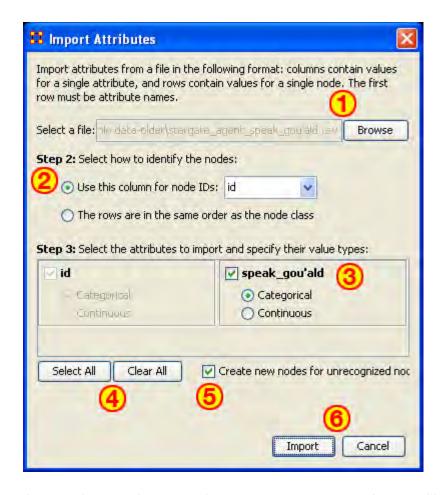
Delete Attribute

And attributes can be removed just as easily by pressing the **Delete** button. The dialog box will present you with a dropdown menu where you can choose which attribute to delete.

Import Attribute

The **[Import]** button allows for the quick importing or an attribute(s) through the Editor panel. Use a **.csv** file with rows as nodes and columns as attributes. This can be done with, or without, an ID column.

1. Use the Textbox and Browse button to navigate to the csv file with the new attribute(s).

2. Select the radio button to use either **1)** an ID column, or **2)** a file in the same order as NodeSet.

3. This section contains all the attributes in the file. If one was chosen to identify the NodeSet then it will be greyed out. For attributes NOT greyed out you can chose to make then **Categorical*** or **Continuous***.

4. **Select All / Clear All** buttons places a checkmark in all columns or removes same.

5. Placing a checkmark in the **Create new nodes for unrecognized nodes** creates new nodes into the select NodeSet if no node for an attribute currently exists.

6. **Import / Cancel** tells ORA to begin the import or cancel the operation.

   Note: Whenever you set a Node Display Attribute for a Nodeset, the info is also set as an attribute of the nodeset. In other words, Node

Display Attributes are now encoded in the data, allowing it to be
retained from session to session.
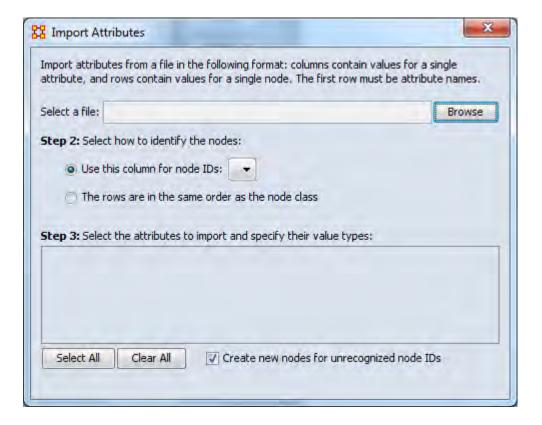
 Attribute Buttons

Attribute Buttons

Used within the Editor to manipulate the attributes of a NodeClass.
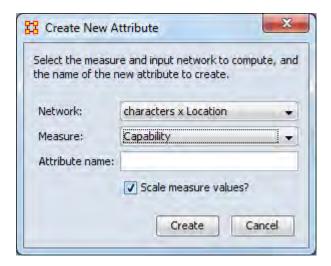
Import : Import attributes from a file in the following format: columns
contain values for a single attribute, and rows contain values for a single
node. The first row must be attribute names. >

Create : You will be asked for the name of the new attribute as well as the Type: **Text, Text Category, URI, Number Category, Number, Date**.

Delete : Ora will ask which attribute you want to delete

Measures : Select the measure and input network to comput and the name of the new attribute to create.
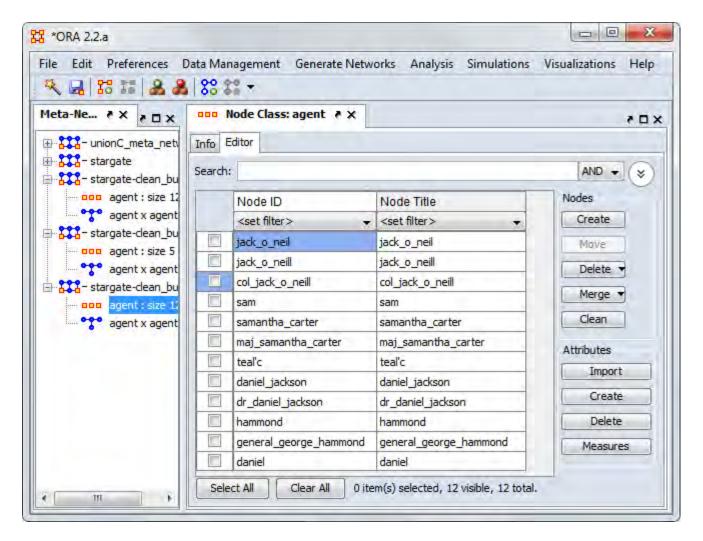
Clean Button
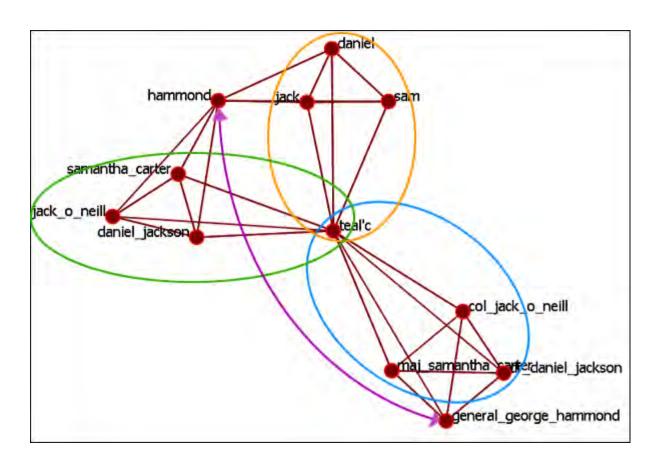
When large datasets are created people's names are after written differently through many different documents. For example you could see someone referred to as **John** in one documents, **John Smith** in another doucments, and **John W. Smith** in yet another document. All three of the these names refer to the same person but in ORA they would be considered different individuals.
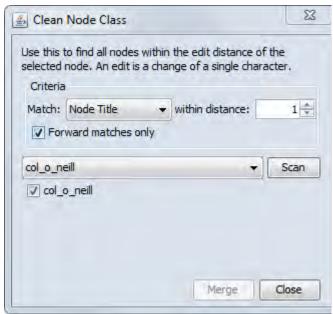
The [**Clean**] function is used to condense all the different referenes of one person into a single node. In the example below there are three instances for Colonel Jack O'Neill (jack_o_neil, jack_o_neill, and col_jack_o_neill), three instances for Major Samantha Carter (sam, samantha_carter, and maj_samantha_carter), Two instance for Doctor Daniel Jackson (daniel_jackson and dr_daniel_jackson) and finally two instance for General George Hammond (hammond and general_george_hammond).

NOTE : Currently this functions searches for changes in the **Node ID** attribute only.
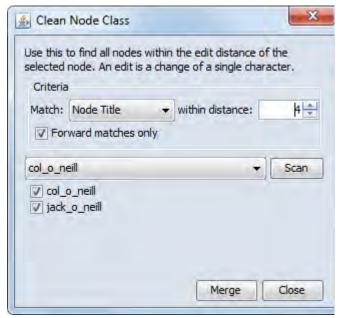
Visualizing this network makes it appear there are quite a few more people there actually are in the network (12 as opposed to 5).

From observing in the Editor there are three instances of the agent **Jack O'Neill**. With the **Min-Edit** distance set to 0 the only agent visible is **jack_o_neil** (with only one l).
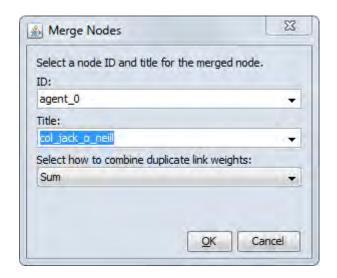
Increasing the Min-Edit to 1 now makes visible a second agent, **jack_o_neill**. The difference between the two agents in the spelling of the surname (one (l) as opposed to two (l)s)
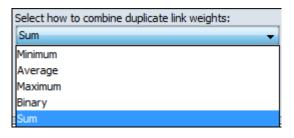
Finaly increasing the Min-Edit to 5 reveals the final agent needed to be merged, **col_jack_o_neill**. The difference in this entry in the inclusion of his rank (col_). The four letter difference here, plus the change from one (l) to two (l)s makes a total of a five point difference.
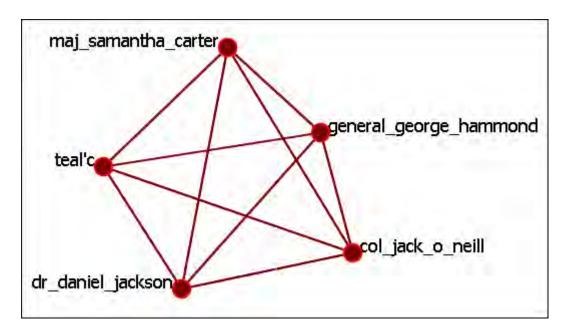
Place a check mark next to all three nodes and click the [**Merge**] button to combine all three nodes into a single node.

Chose one ID: , one Title: , and one method for combining weights of links (as shown below).

After going through the entire Meta-Network and merging all the same people into single nodes I have ended up with a total of five nodes instead of 12 nodes.
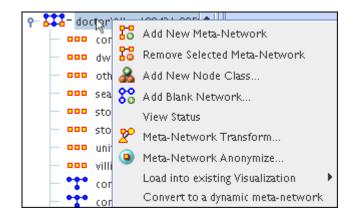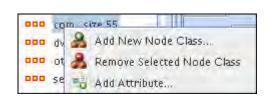
Contextual Menus-Meta-Networks

Right-clicking on any Meta-Network, NodeSet, or Network brings up a contextual menu with the functions available such as add or delete. And within each contextual menu are specialty functions for Meta-Networks, NodeSets, and Networks.
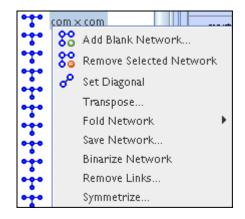
The first four are self-explanatory. **Transform…** opens up the **Meta-Network Transform…** dialog box.
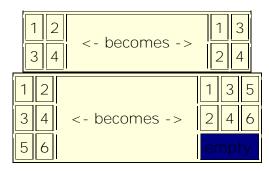
The first two are self-explanatory. The **Add Attribute …** item opens up the **attribute function** in order to add attributes to a nodeset.

Adding and Removing a Network are fairly self-explanatory. Below is a description of the other functions.
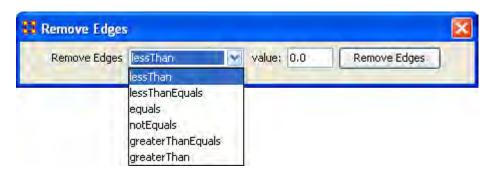
- **Set Diagonal… :** Used on a square network to set cells 1,1 through x,x to the same value. In binary view the choice are **1: True (+1) 2: True (-1) 3: False (0)**. In Numeric view you can put any value into the diagonal.

- **Transpose :** Rotates axis **X** 90 degrees clockwise and mirrors the image horizontally.

| 1 | 2 | <- becomes -> | 1 | 3 |
| 3 | 4 | | 2 | 4 |

1	2	<- becomes ->	1	3	5
3	4		2	4	6
5	6		empty

- **Fold Network... :** This function creates a new network using matrix algebra. Below are four variations of a **four x four** network and the results when each is folded.

	T1	T2
Knowledge_1	1.0	0.0
Knowledge_2	0.0	0.0

→

	Knowledge_1	Knowledge_2
Knowledge_1	1.0	0.0
Knowledge_2	0.0	0.0

	T1	T2
Knowledge_1	1.0	0.0
Knowledge_2	0.0	1.0

→

	T1	T2
Knowledge_1	1.0	0.0
Knowledge_2	0.0	1.0

	T1	T2
Knowledge_1	1.0	0.0
Knowledge_2	1.0	0.0

→

	Knowledge_1	Knowledge_2
Knowledge_1	1.0	1.0
Knowledge_2	1.0	1.0

	T1	T2
Knowledge_1	1.0	1.0
Knowledge_2	0.0	0.0

→

	Knowledge_1	Knowledge_2
Knowledge_1	2.0	0.0
Knowledge_2	0.0	0.0

- **Save Network... : (also called Exporting a Network) Any Network can be saved individually to a file in one of the following formats: CSV*,** DL, **DyNetML*,** or UNCINET (.##h). Also check **File Formats** for more information

- **Binarize Network :** Turns all **non-zero numbers** to **[1]** and leaving all **[0]** untouched.

- **Remove Links... :** Removes links in accordance to the selection in the dropdown menu (as seen in the images below).

- **Symmetrize... :**

🔲 Contextual Menus - Multi Files
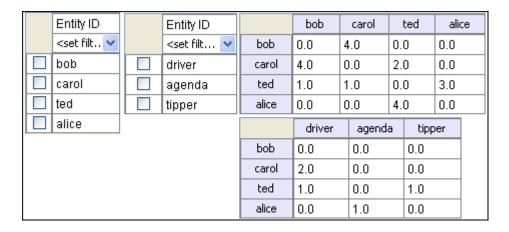
Includes: *Union Meta-Networks, Intersect Meta-Networks, Conform Meta-Networks*

There's also a separate contextual menu when you've got two or more Meta-Networks selected. Open up multiple Meta-Networks in the Main Interface. Load two or more files into Panel 1. We'll demonstrate this on two Meta-Networks containing Bob, Carol Ted, & Alice. The agents are identical but the tasks in each are different, except for **driving** which appears in both and has slightly different values. Inbetween the time of cookingIn and eatingOut Alice's feelings for Ted have grown.

CookingIn and
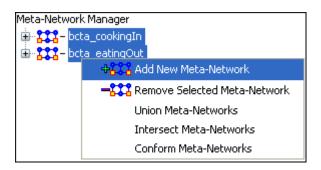
Entity ID		Entity ID							
<set filt.. ⌄		<set filt... ⌄				bob	carol	ted	alice
☐ bob		☐ shopping		bob		0.0	4.0	0.0	0.0
☐ carol		☐ chief_cook		carol		4.0	0.0	2.0	0.0
☐ ted		☐ waiter(ess)		ted		1.0	2.0	0.0	1.0
☐ alice		☐ bottle_wa...		alice		0.0	0.0	4.0	0.0
		☐ busboy(girl)							
		☐ sous_chef							
		☐ driver							

	shopping	chief_cook	waiter(ess)	bottle_washer	busboy(girl)	sous_chef	driver
bob	1.0	1.0	0.0	0.0	0.0	0.0	1.0
carol	0.0	0.0	1.0	1.0	0.0	0.0	1.0
ted	0.0	0.0	0.0	1.0	0.0	1.0	0.0
alice	1.0	0.0	0.0	0.0	1.0	1.0	0.0

EatingOut

Entity ID		Entity ID			bob	carol	ted	alice
<set filt.. ⌄		<set filt... ⌄		bob	0.0	4.0	0.0	0.0
☐ bob		☐ driver		carol	4.0	0.0	2.0	0.0
☐ carol		☐ agenda		ted	1.0	1.0	0.0	3.0
☐ ted		☐ tipper		alice	0.0	0.0	4.0	0.0
☐ alice								

	driver	agenda	tipper
bob	0.0	0.0	0.0
carol	2.0	0.0	0.0
ted	1.0	0.0	1.0
alice	0.0	1.0	0.0

Meta-Network Manager
⊞ – bcta_cookingIn
⊞ – bcta_eatingOut
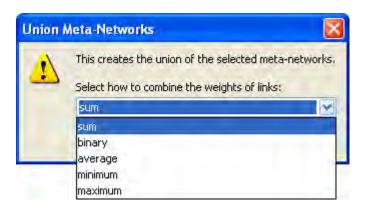
Highlight both of them by holding down the **[Control]** while clicking on each file. Then **Right-click** on one of the files. This brings up the contextual menu.

The **Add New Meta-Network & Remove Selected Meta-Network** are self-explanatory.

The Union Meta-Network will create a **new** Meta-Network using one of five actions: **sum, binary, average, minimum, or maximum**.

- **Sum :** In any identical network **all** values from **all** networks are added together. (i.e. bob's score of 4.0 in both meta-networks are added together for a total of 8.0).

	bob	carol	ted	alice
bob	0.0	8.0	0.0	0.0
carol	8.0	0.0	4.0	0.0
ted	2.0	3.0	0.0	4.0
alice	0.0	0.0	8.0	0.0

- **Binary :** Then in the binary option it doesn't matter what numbers appeared in either meta-network as it uses **only 1 or 0** as a result. If any cell has a **non-zero** it will contain a 1 as a result.

	bob	carol	ted	alice
bob	0.0	(1.0)	0.0	0.0
carol	1.0	0.0	1.0	0.0
ted	1.0	1.0	0.0	1.0
alice	0.0	0.0	1.0	0.0

- **Average :** This option takes the **sum of all identical cell values and divides them by the number of cells used. In cookingInted x alice** contained a 2 while in **eatingOut** the value for **ted x alice** was a 1. This was averaged out to 1.5.

	bob	carol	ted	alice
bob	0.0	4.0	0.0	0.0
carol	4.0	0.0	2.0	0.0
ted	1.0	(1.5)	0.0	2.0
alice	0.0	0.0	4.0	0.0

- **minimum :** This function finds the smallest value in any identical cells and uses that in the final result. (i.e. for **ted x carol** and **ted x alice** both cells use the smaller value of "1" even though they are from different meta-networks.

	bob	carol	ted	alice
bob	0.0	4.0	0.0	0.0
carol	4.0	0.0	2.0	0.0
ted	1.0	(1.0)	0.0	(1.0)
alice	0.0	0.0	4.0	0.0
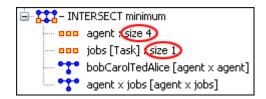
- **Maximum :** This function finds the largest value in any identical cells and uses that in the final result. (i.e. for **ted x carol** and **ted x alice** both cells use the larger value of "2" and "3" respectively, each taken from a different meta-network.

	bob	carol	ted	alice
bob	0.0	4.0	0.0	0.0
carol	4.0	0.0	2.0	0.0
ted	1.0	(2.0)	0.0	(3.0)
alice	0.0	0.0	4.0	0.0

Intersect Meta-Network works similar to the Union function and has the same five options. But in creating the new Meta-Network only nodes that appear in all Meta-Networks are carried over to the new Meta-Network.

For example: All four agents appear in both Meta-Networks and are brought over into the new Meta-Network. But though there were **six tasks** in the

cookingIn and **three tasks** in eatingOut there is only **one task (driver)** in the new Meta-Network created from the intersect function. Only nodes found in all Meta-Networks are brought over.

And even though **bob** has a value for driver in **cookingIn** only carol has a value for driver in both Meta-Networks.

	bob	carol	ted	alice		driver
bob	0.0	4.0	0.0	0.0	bob	0.0
carol	4.0	0.0	2.0	0.0	carol	1.0
ted	1.0	2.0	0.0	1.0	ted	0.0
alice	0.0	0.0	4.0	0.0	alice	0.0

Conform Meta-Network alters the selected Meta-Networks and makes them equal. **union** adds nodes that are found in one Meta-Network but not the other. **intersect** removes nodes that are not common to both.

Contextual Menus-Networks

Right-clicking on any network brings up the contextual menu with a set of functions for working with Networks.

Hide selected

Select one or more rows or columns. Then right-click to bring up the contextual menu. Selecting **Hide Selected** will remove the selected rows or columns from the display.

NOTE : They are hidden for as long as you have the current network viewed. Clicking anywhere else in the Editor then returning back to the previous Network will reveal all rows or columns which were hidden.

Sort

The order which nodes are displayed is the default order they were entered. ORA gives you four different ways to display these same nodes.

- **Ascending by Title A↓Z :** Arranges nodes in alpha order lowest to highest.

- **Ascending by Link Count :** Arranges nodes in order of number of links from lowest to highest.

- **Descending by Title Z↓A :** Arranges nodes in alpha order highest to lowest.

- **Descending by Link Count :** Arranges nodes in order of number of links from highest to lowest.

Ascending by Title
Ascending by Link count
Descending by Title
Descending by Link count

NOTE : Nodes will retain their sort order for the duration they are displayed on the screen. Changing this view and returning to a network will place the nodes back in their original order.

Hightlight

Highlights or clears selected nodes.

- **Clear all :** Clears the highlight from all nodes.

- **Clear selected :** Clears the highlight from only the selected nodes.

- **Hightlight selected :** Highlights all nodes currently selected.

Clear all
Clear selected
Highlight selected

Example
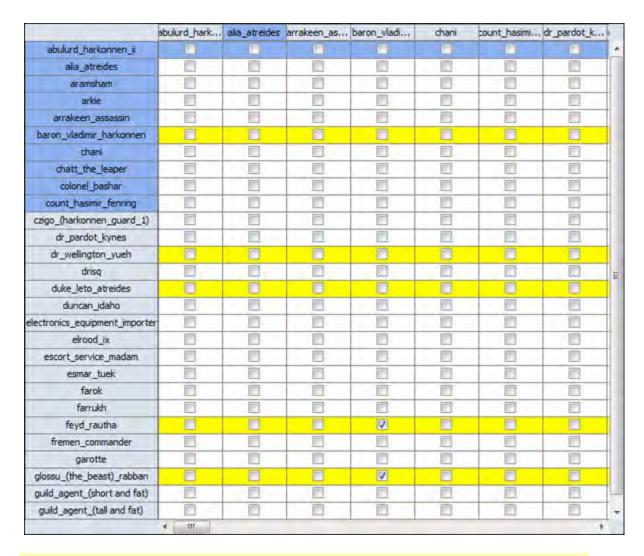
First I sorted the agents by [**Right-Clicking**] on any agent and selecting **Sort > Descending by link count** so the nodes with the highest numbers of links would rise to the top. I clicked on **paul_atreides**, counted down ten agents, and [**Shift-Clicked**] on **thufor_hawat**. This selected the ten agents with the most links.

	abulurd_hark...	alia_atreides	arrakeen_as...	baron_vladi...	chani
paul_atreides	☐	☐	☐	☐	☐
baron_vladimir_harkonnen	☐	☐	☐	☐	☐
duke_leto_atreides	☐	☐	☐	☐	☐
lady_jessica	☐	☐	☐	☐	☐
dr_wellington_yueh	☐	☐	☐	☐	☐
feyd_rautha	☐	☐	☐	☑	☐
glossu_(the_beast)_rabban	☐	☐	☐	☑	☐
piter_de_vries	☐	☐	☐	☑	☐
reverand_mother_gaius_hele...	☐	☐	☐	☐	☐
thufir_hawat	☐	☐	☐	☐	☐
duncan_idaho	☐	☐	☐	☐	☐
harah_(jamis_wife)	☐	☐	☐	☐	☐

Hovering over any of the selected agents I [**Right-Clicked**] bringing up the contextual menu. I select **Hightlight > Highlight selected**. This placed a yellow highlight over the top ten agents.

	abulurd_hark...	alia_atreides	arrakeen_as...	baron_vladi...	chani
paul_atreides	☐	☐	☐	☐	☐
baron_vladimir_harkonnen	☐	☐	☐	☐	☐
duke_leto_atreides	☐	☐	☐	☐	☐
lady_jessica	☐	☐	☐	☐	☐
dr_wellington_yueh	☐	☐	☐	☐	☐
feyd_rautha	☐	☐	☐	☑	☐
glossu_(the_beast)_rabban	☐	☐	☐	☑	☐
piter_de_vries	☐	☐	☐	☑	☐
reverand_mother_gaius_hele...	☐	☐	☐	☐	☐
thufir_hawat	☐	☐	☐	☐	☐
duncan_idaho	☐	☐	☐	☐	☐
harah_(jamis_wife)	☐	☐	☐	☐	☐

I wanted the list sorted by name again so I [**Right-Clicked**] and selected **Sort > Ascending by title.**

	abulurd_hark...	alia_atreides	arrakeen_as...	baron_vladi...	chani	count_hasimi...	dr_pardot_k...
abulurd_harkonnen_ii	☐	☐	☐	☐	☐	☐	☐
alia_atreides	☐	☐	☐	☐	☐	☐	☐
aramsham	☐	☐	☐	☐	☐	☐	☐
arkie	☐	☐	☐	☐	☐	☐	☐
arrakeen_assassin	☐	☐	☐	☐	☐	☐	☐
baron_vladimir_harkonnen	☐	☐	☐	☐	☐	☐	☐
chani	☐	☐	☐	☐	☐	☐	☐
chatt_the_leaper	☐	☐	☐	☐	☐	☐	☐
colonel_bashar	☐	☐	☐	☐	☐	☐	☐
count_hasimir_fenring	☐	☐	☐	☐	☐	☐	☐
czigo_(harkonnen_guard_1)	☐	☐	☐	☐	☐	☐	☐
dr_pardot_kynes	☐	☐	☐	☐	☐	☐	☐
dr_wellington_yueh	☐	☐	☐	☐	☐	☐	☐
drisq	☐	☐	☐	☐	☐	☐	☐
duke_leto_atreides	☐	☐	☐	☐	☐	☐	☐
duncan_idaho	☐	☐	☐	☐	☐	☐	☐
electronics_equipment_importer	☐	☐	☐	☐	☐	☐	☐
elrood_ix	☐	☐	☐	☐	☐	☐	☐
escort_service_madam	☐	☐	☐	☐	☐	☐	☐
esmar_tuek	☐	☐	☐	☐	☐	☐	☐
farok	☐	☐	☐	☐	☐	☐	☐
farrukh	☐	☐	☐	☐	☐	☐	☐
feyd_rautha	☐	☐	☐	☑	☐	☐	☐
fremen_commander	☐	☐	☐	☐	☐	☐	☐
garotte	☐	☐	☐	☐	☐	☐	☐
glossu_(the_beast)_rabban	☐	☐	☐	☑	☐	☐	☐
guild_agent_(short and fat)	☐	☐	☐	☐	☐	☐	☐
guild_agent_(tall and fat)	☐	☐	☐	☐	☐	☐	☐

NOTE : Highlighting works in the same manner with regards to columns.

13 JAN 11

Duplicate Items

ORA allows for the quick duplication of Meta-Networks, Nodesets, and Networks. It uses the **Copy and Paste** commands but some aspects work in a different manner than most users are familiar with.

Meta-Network

Duplicating a Meta-Network can be done with the menu, **Edit > Copy Ctrl+C** then **Edit > Paste Ctrl+V**. This will create a duplicate of the highlighted Meta-Network. And as the menu items show you can also use the keyboard shortcuts, **Crtl+C and Ctrl+V**

62

There is also a method for duplicating two or more Meta-Networks simultaneously. Hightlight all the Meta-Networks you need to duplicate. Press and hold the [**Alt**] key. Place the mouse cursor over one of the highlighted Meta-Networks then **Click and hold**. Drag the cursor into an unused area of the Meta-Network Manager pane and release the mouse button. All highlighted Meta-Networks will be duplicated.

NodeSets and Networks

Using the Menu items **Edit > Copy Ctrl+C** and **Edit > Paste Ctrl+V** work differently with NodeClasses and Networks. This can be used to either create new Meta-Networks or duplicate ModeClasses and Networks in existing Meta-Networks.

This leads to an interesting side benefit. You can highlight NodeClasses and Networks from different Meta-Networks, Copy, and Paste. This will create a new Meta-Network with all the highlighted items.

Create new Meta-Network : As with Meta-Networks click, hold and drag a NodeClass(es) or Network(s) to an unused space in Meta-Netowrk Manager pane. Upon releasing the mouse button ORA will create new Meta-Networks containing those NodeClasses and Networks.

NOTE : If you highlight two NodeClasses or Networks with the same name ORA will ask you to give the second one a different name.

Add to existing Meta-Network : Dragging a NodeClass or Network to an existing Meta-Network will add that NodeClass or Network to the new Meta-Network. If that NodeClass name is currently in use ORA will as you to rename the NodeCLass or Network you want to append.

09 MAR 11

Create a New Meta-Network

What follows are procedures for creating a new Meta-Network.

There are times you need to run measures on or visualize only part of an existing Meta-Network. This section will show you the easy way to create sub-sets of an existing Meta-Network.
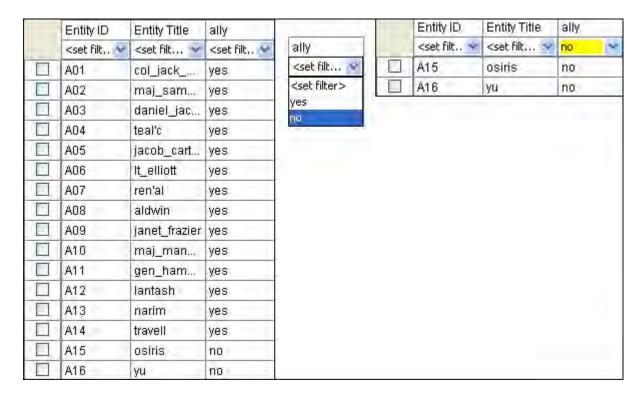
First **duplicate the Meta-Network**. Highlight the Meta-Network then from the main menu select **Edit ⇒ Copy Meta-Network**. Then from the main menu select **Edit ⇒ Paste Meta-Networks**. It would be a good idea to

rename this new Meta-Network immediately so as to not get it confused with the original. You can do this in the Editor tab, see **Renaming**.

Next, expand the Meta-Network you want by clicking the **[+]** if it isn't expanded already. Highlight the nodeset with the nodes you want to remove.

In panel 2 select the **Editor tab** which brings up the matrix with nodes on the vertical and attributes on the horizontal. Below each attribute is a **<set filter>**. Below left is the **stargate_summit** meta-network along with the **ally** attribute. I wish to make a new meta-network of the **good guys** only. In the dropdown menu under ally I select **no**. These are the nodes I want to remove.

Place a checkmark in the box of each node. Below the matrix in the section called **Nodes** are three buttons. Click the **[Delete]** button. The nodeset now contains only the nodes that had **yes** in the attribute **ally**.
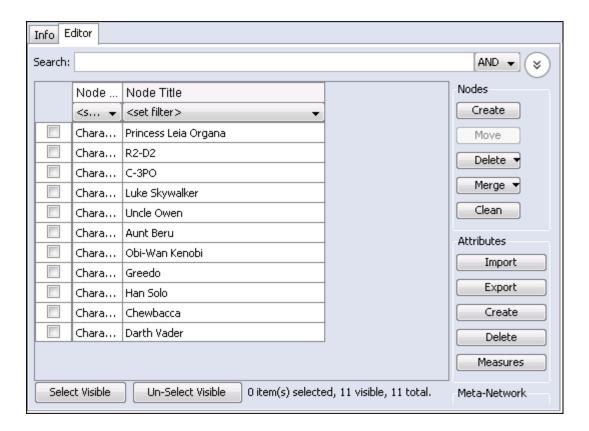
With this same procedure I created a Meta-Network of only the **allies**.

Create New Node

With the [**Create**] button in the editor you can create any number of nodes at one time. But there are times you only need to insert a sigle node into a nodeset. Such as while building a dataset you forgot to include it in the original construction. ORA gives you a quick method of adding a single node.
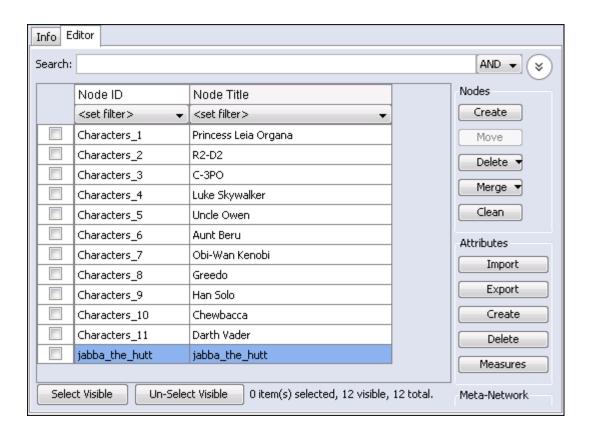
Below is the small Star Wars set containing 11 agents. And for some reason you needed to include a 12th agent, say Jabba the Hutt.

I typed **jabba_the_hutt** into the Search box and since he's not in this set ORA display no nodes. But notice to the right of the Search box a [**Create a new node**] button has appeared. If the node you want does not exist this allows you to instantly add it to the nodeset.
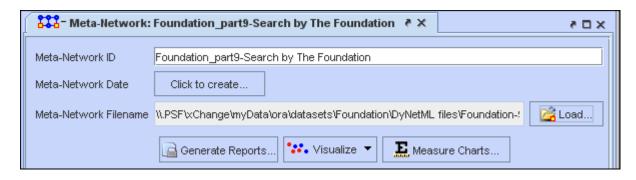
Upon click the [**Create a new node**] button the Jabb_the_hutt node is added to the bottom of the list.

12 JAN 11

Dating Meta-Networks
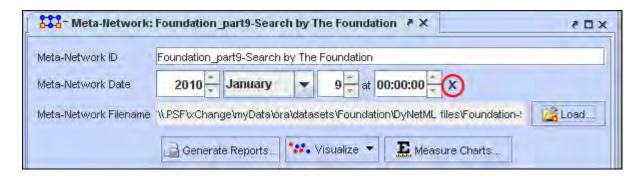
Certain functions require a Meta-Network to have a date attached to it. These include: Measures Over Time, Networks Over Time, and Loom. To add a date first highlight the Meta-Network. In the Editor pane find **Meta-Network Date**. Click on the [**Click to create...**].

Now you can set the parameters for the date which include year, month, day, and time.

Remove Meta-Network date

If, at any time, you need to remove the date from a Meta-Network click on the [**X**] to the right of the date.

 Use Links As Attributes

The following instruction demonstrates how to add a **document attribute** or a **URI** to a Node.

Adding a document links in the Editor

Load a Meta-Network into ORA. Then expand the Meta-Network using the [+] icon to display all the NodeClasses and Networks in the Meta-Network. Then right-click on a NodeClass and this will access the contextual menu. From here, select **Add Attribute**. Give the new attribute a name.

In Pane 2 click the Editor tab to display the matrix and find the new attribute just created. All cells will be empty. A new pathway can be inserted in the cell at this time.

Once the node title has been changed, you can now enter a URI directly inside the corresponding blank field. This can either be an absolute pathway to a document on the user's hard drive or a web URL.

Adding a URI in the Visualizer

Visualize a Meta-Network, select a node, and right-click to access it's contextual menu. From the contextual menu select, choose **URI submenu > Attach URI**. This brings up the **Add URI** box where the new URI can be entered.

When viewing the Meta-Network in the Editor a new column appears with the title you inserted in the field **Name:**.

The same document pathway can be used for multiple nodes.

When the Meta-Network is displayed in the Visualizer the document URI will appear in the Node Status tooltip. You can go to this URI by right-clicking a node and use the URI submenu to **Go To URI**. A new explorer window will be launched showing the location of the document file. It can then be opened with the appropriate program.

Multiple URIs/URLs

A node can have multiple URIs and you can still launch from the URI SubMenu. The preferred method for launching URIs is through the **Node Status** dialog where you can choose an individual URI to launch.

There's a new URI tab that allows you to view URI's, launch them, and add new ones. When adding new ones, the current convention is to create a new Attribute with name URI-X, where X is the earliest number that doesn't already exist. So, adding a URI to nodes will create results such as URI-1, URI-2, etc. Currently, when finding URI's to \launch, only those that follow the above convention are included, but it will be expanded to include any attribute with URI or URL in it anywhere.

Editor Toolbar

To help facilitate your workflow ORA allows you to mount/unmount tool buttons in the Editor's toolbar (and also in the Visualizer toolbar). Click the down-triangle to open the dialog box.

The **Configure Toolbar** dialog box will appear. You will notice that some items are already checked. These correspond to the buttons already present. Placing a check mark next to any tool instantly mounts it on the toolbar. Removing the check mark from a button instantly removes the

button.

When you are finished click [**Close**].

If at any point you can select the [**Restore Default**] button which will remove all the changes you made and restore the toolbar to the buttons found in the image at the top.

NOTE : The buttons appear from left to right in the order you select them. You can not re-arrange buttons once they are placed on the toolbar.

 # Filters

Use of this function allows you to reduce the number of nodes that appear in the Editor. By clicking the double up-pointing arrows in the top right of the Editor pane you will reveal the filtering tabs. Using the [**+**] you can add additional filtering parameters. Likewise you can remove filters by using the **[X]**

General Filter

The General Filter works across all columns. Think of it as a **blunt force** filter. The terms you are searching for are either **In or are not in** the dataset.

Attribute Filter

With the Attribute Filter you can do finely tuned searches specifying whether a term is foun, or not, within a specific attribute.

NOTE : At any point you need to backtrack or start over select the **General Filter** tab and click the [**Reset**] button.
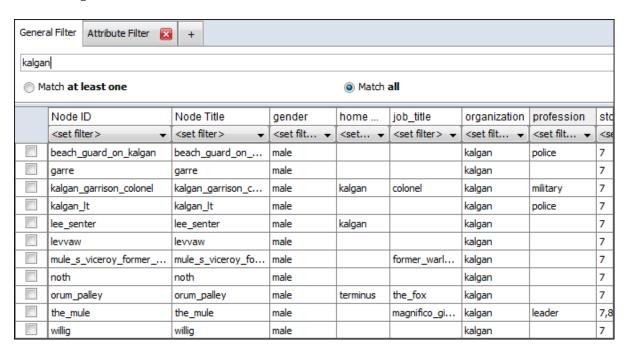
General Filtering

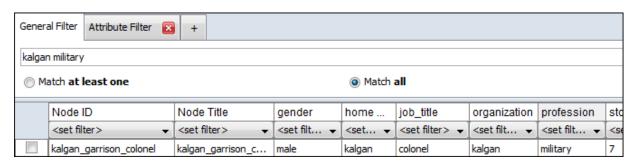Here is the character nodeset from part 7 of the Foundation dataset.

Node ID	Node Title	gender	home ...	job_title	organization	profession
<set filter> ▼	<set filter> ▼	<set filt... ▼	<set... ▼	<set filter> ▼	<set filt... ▼	<set filt... ▼
anonymous_conspirator	anonymous_consp...	male	terminus	worker		spy
bayta_darell	bayta_darell	female	terminus		foundation,...	spy
beach_guard_on_kalgan	beach_guard_on_...	male			kalgan	police
bureau_chief_of_produc...	bureau_chief_of_...	male	haven	bureau_chief	bureaucrat	bureaucrat
captain_han_pritcher	captain_han_pritcher	male	locris	war_depart...	foundation,...	military
captain_inspector_of_fili...	captain_inspector...	male				military
dagobert_IX	dagobert_IX	male	trantor	emperor	empire	royalty
ebling_mis	ebling_mis	male		the_scientist	foundation	psychologist
filian_soldier_1	filian_soldier_1	male				military
filian_soldier_2	filian_soldier_2	male				military
filian_soldier_3	filian_soldier_3	male				military
filian_soldier_4	filian_soldier_4	male				military
filian_soldier_5	filian_soldier_5	male				military
filian_soldier_6	filian_soldier_6	male				military
foundation_tax_collector	foundation_tax_c...		terminus	tax_collector	foundation	finance
franssart	franssart	male		toran_s_father	trader	trader
garre	garre	male			kalgan	
general_sammin	general_sammin	male		general		military
gilmer	gilmer					
governor_jord_commason	governor_jord_co...	male		governor	empire	bureaucrat
hella	hella	female	haven	worker	trader	worker
inchney_(dagobert_X)	inchney_(dagober...	male	neotrantor	crown_prince	empire	royalty
indbur_s_chamberlain	indbur_s_chamberl...	male	terminus		faoundation	servant
iwo_lyon	iwo_lyon	male			trader	trader
juddee	juddee	female	haven	worker	trader	worker
kalgan_garrison_colonel	kalgan_garrison_c...	male	kalgan	colonel	kalgan	military
kalgan_lt	kalgan_lt	male			kalgan	police
lee_senter	lee_senter	male	kalgan		kalgan	
levvaw	levvaw	male			kalgan	
mangin	mangin	male	iss		trader	trader
mayor_indbur_the_third	mayor_indbur_the...	male	terminus	mayor	foundation	politician
mule_s_viceroy_former_...	mule_s_viceroy_fo...	male		former_warl...	kalgan	
new_girl	new_girl	female	haven	worker	trader	worker
noth	noth	male			kalgan	
orum_palley	orum_palley	male	terminus	the_fox	kalgan	
ovall_gri	ovall_gri	male	mnemon		trader	trader
porfirat_hart	porfirat_hart				foundation,...	spy
randu	randu	male		toran_s_fat...	trader	trader
secretary_to_the_mayor	secretary_to_the_...	male	terminus	secretary	foundation	civil_servant
the_mule	the_mule	male		magnifico_gi...	kalgan	leader
toran_darell	toran_darell	male	haven		trader	trader

Using the General Filter on the Foundation dataset we'll use it to find the military officers on the world Kalgan. Note that Kalgan is in the attribute HomeWorld and police is in the attribute profession. After typing **kalgan**
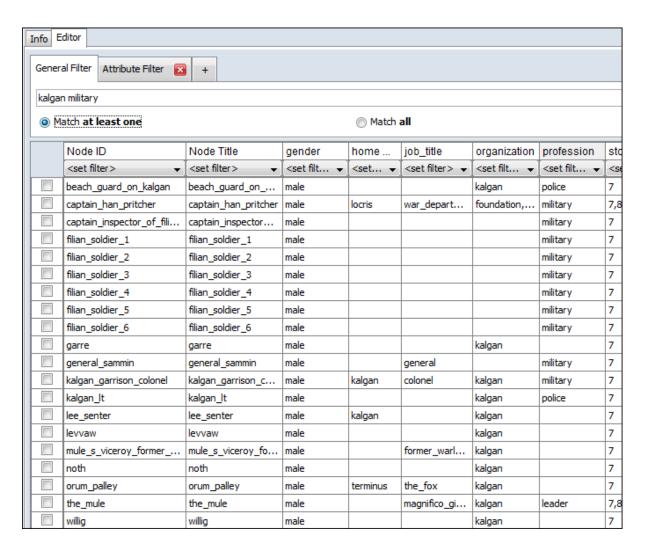
into the search box ORA filters out all entries which **do not** contain the world kalgan.

	Node ID	Node Title	gender	home ...	job_title	organization	profession	std
	<set filter> ▼	<set filter> ▼	<set filt... ▼	<set... ▼	<set filter> ▼	<set filt... ▼	<set filt... ▼	<se
☐	beach_guard_on_kalgan	beach_guard_on_...	male			kalgan	police	7
☐	garre	garre	male			kalgan		7
☐	kalgan_garrison_colonel	kalgan_garrison_c...	male	kalgan	colonel	kalgan	military	7
☐	kalgan_lt	kalgan_lt	male			kalgan	police	7
☐	lee_senter	lee_senter	male	kalgan		kalgan		7
☐	levvaw	levvaw	male			kalgan		7
☐	mule_s_viceroy_former_...	mule_s_viceroy_fo...	male		former_warl...	kalgan		7
☐	noth	noth	male			kalgan		7
☐	orum_palley	orum_palley	male	terminus	the_fox	kalgan		7
☐	the_mule	the_mule	male		magnifico_gi...	kalgan	leader	7,8
☐	willig	willig	male			kalgan		7

Next **military** is typed into the search box. Remember to leave a space in-between each word. And since **Match all** is selected ORA filters out all entries with do not include **BOTH** kalgan and military. Only one entry corresponds to both.

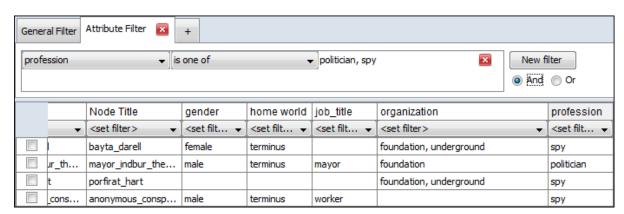|	Node ID	Node Title	gender	home ...	job_title	organization	profession	std
	<set filter> ▼	<set filter> ▼	<set filt... ▼	<set... ▼	<set filter> ▼	<set filt... ▼	<set filt... ▼	<se
☐	kalgan_garrison_colonel	kalgan_garrison_c...	male	kalgan	colonel	kalgan	military	7

Now let's change the conditions from **Match all** to **Match at least one**. Now every entry which has **either** kalgan -OR- military shows up.
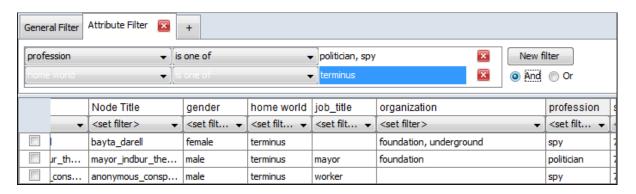
First screenshot (General Filter tab):

Tab bar: Info | Editor

Sub-tabs: General Filter | Attribute Filter [x] | +

Filter textbox: kalgan military

(•) Match **at least one**   ( ) Match **all**

	Node ID	Node Title	gender	home ...	job_title	organization	profession	sto
	\<set filter\> ▼	\<set filter\> ▼	\<set filt... ▼	\<set... ▼	\<set filter\> ▼	\<set filt... ▼	\<set filt... ▼	\<se
☐	beach_guard_on_kalgan	beach_guard_on_...	male			kalgan	police	7
☐	captain_han_pritcher	captain_han_pritcher	male	locris	war_depart...	foundation,...	military	7,8
☐	captain_inspector_of_fli...	captain_inspector...	male				military	7
☐	filian_soldier_1	filian_soldier_1	male				military	7
☐	filian_soldier_2	filian_soldier_2	male				military	7
☐	filian_soldier_3	filian_soldier_3	male				military	7
☐	filian_soldier_4	filian_soldier_4	male				military	7
☐	filian_soldier_5	filian_soldier_5	male				military	7
☐	filian_soldier_6	filian_soldier_6	male				military	7
☐	garre	garre	male			kalgan		7
☐	general_sammin	general_sammin	male		general		military	7
☐	kalgan_garrison_colonel	kalgan_garrison_c...	male	kalgan	colonel	kalgan	military	7
☐	kalgan_lt	kalgan_lt	male			kalgan	police	7
☐	lee_senter	lee_senter	male	kalgan		kalgan		7
☐	levvaw	levvaw	male			kalgan		7
☐	mule_s_viceroy_former_...	mule_s_viceroy_fo...	male		former_warl...	kalgan		7
☐	noth	noth	male			kalgan		7
☐	orum_palley	orum_palley	male	terminus	the_fox	kalgan		7
☐	the_mule	the_mule	male		magnifico_gi...	kalgan	leader	7,8
☐	willig	willig	male			kalgan		7

Attribute Filtering

Single Attribute Filter : Click the leftmost drop down and select **profession**. Click the middle drop down and select **is one of**. Type **politician or spy** into the textbox. Four characters fit this filter and are displayed in the Editor.
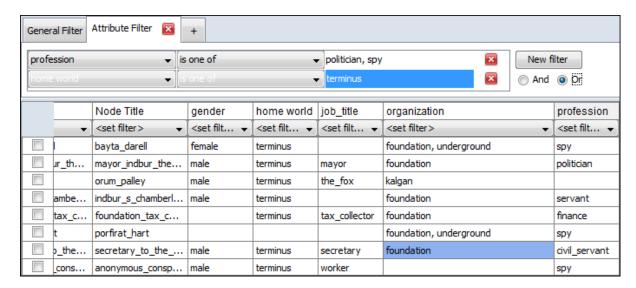

Second screenshot (Attribute Filter tab):

Sub-tabs: General Filter | Attribute Filter [x] | +

profession ▼   is one of ▼   politician, spy   [x]   New filter

(•) And   ( ) Or

		Node Title	gender	home world	job_title	organization	profession
	▼	\<set filter\> ▼	\<set filt... ▼	\<set filt... ▼	\<set filt... ▼	\<set filter\> ▼	\<set filt... ▼
☐	l	bayta_darell	female	terminus		foundation, underground	spy
☐	ur_th...	mayor_indbur_the...	male	terminus	mayor	foundation	politician
☐	t	porfirat_hart				foundation, underground	spy
☐	_cons...	anonymous_consp...	male	terminus	worker		spy

74

Multiple Filters using AND : The search can be further refined by adding additional parameters. Click the [**New filter**] button to create a second filter parameter. It will appear directly beneath your first filter. Click the leftmost drop down and select **home world**. Click the middle drop down and select **is one of**. Type **terminus** into the textbox. This eliminates the character **portirat_hart** from the search as there is no home world specified for that character.

		Node Title	gender	home world	job_title	organization	profession	s
		<set filter>	<set filt...	<set filt...	<set filt...	<set filter>	<set filt...	
	l	bayta_darell	female	terminus		foundation, underground	spy	
	ur_th...	mayor_indbur_the...	male	terminus	mayor	foundation	politician	
	_cons...	anonymous_consp...	male	terminus	worker		spy	

Multiple Filters using OR :

Leaving the filter parameters the same we'll now change the condition from **AND** to **OR**. This means that a character will be displayed in the Editor if **either** the profession or home world match.

		Node Title	gender	home world	job_title	organization	profession
		<set filter>	<set filt...	<set filt...	<set filt...	<set filter>	<set filt...
	l	bayta_darell	female	terminus		foundation, underground	spy
	ur_th...	mayor_indbur_the...	male	terminus	mayor	foundation	politician
		orum_palley	male	terminus	the_fox	kalgan	
	ambe...	indbur_s_chamberl...	male	terminus		foundation	servant
	tax_c...	foundation_tax_c...		terminus	tax_collector	foundation	finance
	t	porfirat_hart				foundation, underground	spy
	_the...	secretary_to_the_...	male	terminus	secretary	foundation	civil_servant
	_cons...	anonymous_consp...	male	terminus	worker		spy

Hovering - Tooltips

Hovering the pointer over various parts of the panels will reveal information about the Meta-Network.

Hovering over the parts in Panel 1 will reveal different information about the **Meta-Networks, Meta-Nodes, and Networks**.

Hovering over the Meta-Network reveals the name.

Hovering over a NodeClass reveals the nodeclass anme and size.

Hovering over a network reveals the network type, it's ID, and the link count.
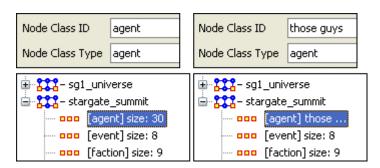
ID Names

Working with ID Names

Each Meta-Network, NodeClass, and Network can be given a distinct **ID Name**. At times this can be very helpful when working with multiple Meta-Networks it can also cause problems with certain functions.

Here I renamed one of the Meta-Nodes to **those guys**. It is also reflected throughout all of Meta-Network Panel. You can rename any Meta-Network, NodeClass, or Network for better readability.

Areas where it's important to maintain Name ID integrity

View Measures Over Time : This function allows an analyst to see how a network measures change over multiple time periods. So in order to track measures over time it is necessary that the ID Names of the Networks are the same in all instances.

View Networks Over Time : This function allows you to set up multiple Networks that encompass various time frames and view them in succession. Since you want to see what happens over the same Network Over Time it becomes virtually important to make sure the ID Names of these Networks are all the same. Else ORA will not be able to analyze them correctly.

Comparing Networks : When ORA runs measures on Networks it's sometimes necessary to do have identical Network names. This will be listed in the Measures section.

Create Meta-Network from Excel File

If you don't have a Network, you can create one from scratch. Below is step-by-step instruction on how to do this in ORA.

We will create an square, agent-by-agent Network. We say it is **square** because all row headings correspond directly to column headings. This is important as it relates to specific measures ORA can run on a graph. If the graph is not square, some measures will not work.

Open a blank Microsoft Excel work book. In **column A** we will enter the name of all the nodes that make up our social network or organization.

NOTE : When creating your spreadsheet, do not add any additional titles, notes, or other headings, which will interfere with the "square" properties of the Network.

◇	A	B	C	D	E
1					
2	bob				
3	carol				
4	ted				
5	alice				

Next, create column headings using the correlating names as they appear in row headings. Again, this will ensure that our Network will be square.

◇	A	B	C	D	E
1		bob	carol	ted	alice
2	bob				
3	carol				
4	ted				
5	alice				

Next we will create **links** between each agent. We do this by entering a **1** if a direct connection or relationship exists and a **0** if it does not. Please note that headings that cross-reference themselves are considered redundant and thus are left blank or **0**.

In the example below, **Redundant cells** are filled in with red strips to illustrate the self-loops. This redundancy should continue as a smooth diagonal line from the top left corner of your Network to the bottom right.

◇	A	B	C	D	E
1		bob	carol	ted	alice
2	bob	▨			
3	carol		▨		
4	ted			▨	
5	alice				▨

NOTE : If you don't end up with a diagonal line then your graph is not square.

Using **1**s and **0**s to establish link, complete your spreadsheet.

In the Network example, we have assigned links randomly. Within your organization or network, however, you can describe any direct connections or relationships you are interested in analyzing. For instance, you may determine that a direct connection exists if agents within your network consult with each other at least once a month; literally, it can be anything you decide.

Below is our completed Network (The red fill illustrates cells that do not require input due to their **redundancy***).

◇	A	B	C	D	E
1		bob	carol	ted	alice
2	bob	▨	1	0	0
3	carol	1	▨	1	0
4	ted	1	1	▨	1
5	alice	0	0	1	▨
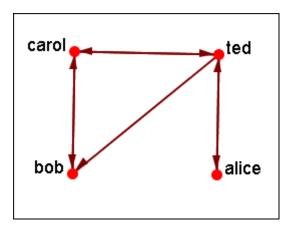
Now that we have essentially built a Network from scratch using Excel, the next step is to save it in a compatible file format ORA can interpret. For Excel spreadsheets this will be the **CSV*** file format.

From the main menu select **File ⇒ Save As**

Make sure you save this file as a **CSV (comma delimited)**

You have now created a Network from scratch which can be loaded into ORA.
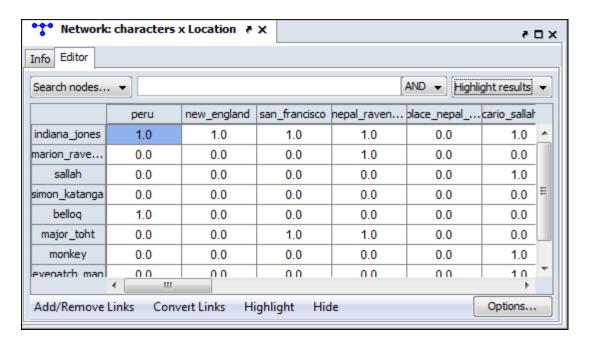
Now return to ORA and load up your new Network. Below is a our new Network rendered in the ORA Visualizer. Notice the arrows only point from one node to another if there is a 1 in the column for a particular node. **i.e. There is a "1" in the bob column for ted but a "0" in the ted column for bob. So an arrow points from ted to bob but NOT from bob to ted**.
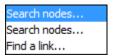
For a more in-depth explanation for creating Meta-Networks see the **lessons section under Meta-Network**.

Network Editor

With a network highlight and the Editor tab displayed ORA has many functions for working with nodes and links in a network.
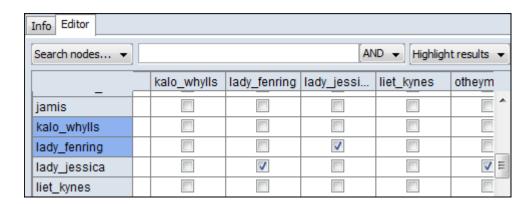
Search Nodes…
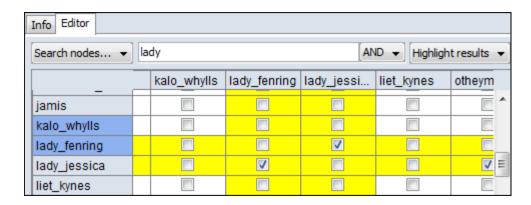
Search type 1
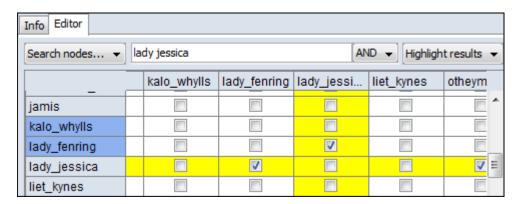
Using the Search box and the [**AND/OR**] selector you can refine a search on a Network.
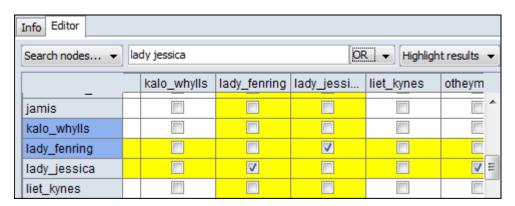
With the [**AND/OR**] selector set to AND I type **lady** into the Search box. This results in two rows/columns being highlighted, **lady_jessica** and **lady_fenring**.

To refine the search I type **jessica** into the Search box (making sure to place a space between the words). This eliminates **lady_fenring** leaving only **lady_jessica** highlighted.

Without changing anything in the Search box I change the **AND** to **OR** in the [**AND/OR**] selector and it now reveals any node with either **lady** OR **jessica**.

NOTE : Idential search results were achieved with different parameters.
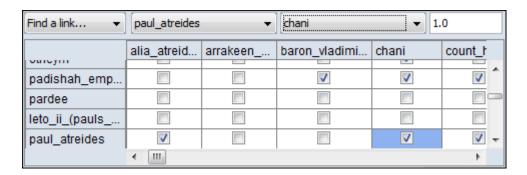
Search type 2

This search is similar as the primary search but with two Search boxes for added refinement.
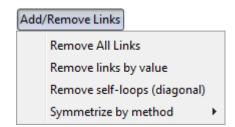
Find a link…

Chosing two nodes in the two dropdown menus will locate the link in the matrix.

Remove All Links : Removes all links from Network.

Remove links by value : Allows you to set parameters for the values of links which you want to remove.

Remove self-loops (diagonel) : Removes links connecting a node to itself/

Symmetrize by method : Gives for options : Maximum, Minimum, Sum, and Average.

Convert Links

Binarize link weights ($x \neq 0 \Rightarrow x = 1$)

Collapse link weights ($a \leq x \leq b \Rightarrow x = 1$)

Negate link weights (-x)

Invert the link weights (1/x)

Logorithm of the link weights (log10(x))

Absolute value of the link weigths (|x|)

Scale the link weights (c * x)

Row-Normalize the link weights (x/OutDegree)

Increment the link weights (c + x)

Convert Links : Gives options for changing the value of links in a Network.

Highlight
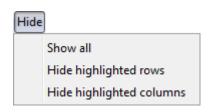
Clear all

Highlight isolates

Clear all : Clears all highlighting from rows and columns.

Highlight isolates : Highlights rows and columns of any node with has no connections to any other nodes in the network.
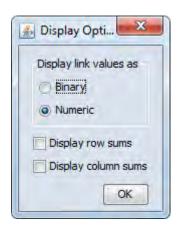
	arrakeen_	baron_vladimi	chani	count_has	dr_pardot	dr_welling	duke_leto	duncan_id	elrood_ix	feyd_rautha	garotte	glossu_it	guildsman	gurney_ha	hu
arrakeen_assa	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
baron_vladimir	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	1.0	0.0	0.0	
chani	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
count_hasimir	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
dr_pardot_kynes	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
dr_wellington_	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
duke_leto_atre	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	1.0	
duncan_idaho	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	
elrood_ix	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
feyd_rautha	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	
garotte	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
glossu_ithe_b	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	
guildsman	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
gurney_halleck	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	
haran_garnis_	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
harkonnen_co	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
jamis	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
kalo_whylis	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
lady_fenring	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
lady_jessica	0.0	0.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
liet_kynes	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	
othoym	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
padishah_emp	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
pardee	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
leto_ii_paul_	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
paul_atreides	0.0	0.0	0.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	
piter_de_vries	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
princess_irulan	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
reverand_moth	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
reverand_moth	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
shadout_mapes	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	
spacing_guild	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
stilgar	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	1.0	
thufir_hawat	0.0	0.0	0.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	

Show all :

Hide highlighted rows :

Hide highlighted columns :

 # Node Buttons

Node Buttons

Used within the Editor you can manipulate nodes in a NodeClass.

NOTE : Any changes done to a NodeClass are permanent. There is no Undo.

Create : inserts a new node(s) into the NodeClass. You will be asked for the number of new nodes to create and a node ID. If multiple nodes are to created then a number will be appended to each new node.

Move : Move all check marked node(s) to another designated NodeClass. ORA will prompt you for the movement of links associated with these node(s).

Delete : Removes node(s) and all associated link(s) from the Meta-Network. There are four sub-options viewable by clicking the downward triangle. The nodes deleted are kept in a log which is used for creating a delete list.

Delete selected nodes : Instantly deletes all nodes which have a check mark placed to the left of their name.

Clear delete list : Clears from the log any information about nodes that were deleted.

Load and apply delete list : You can load a previously saved delete list and ORA will then apply it to the currently selected Meta-Network.

Save delete list : Saves the log of deleted nodes to a file which you can give a distinctive name. Nodes from any NodeClass which were deleted are saved to this list.

Merge : removes all selected nodes and replaces them with a single node with all previous links and attributes. This is used primarily when data needs to be consolidated. **Example :** A NodeClass contains two nodes, **jack** and **col_o'neill**. These are actually the same person. Merge could combine their information into one node which could be named **col_jack_o'neill**.

Merge selected nodes : Merge all check marked nodes into a single node.

Clear merge list : Clears from the log any information about nodes that were merged.

Load and apply merge list : You can load a previously saved merge list and ORA will then apply it to the currently selected Meta-Network.

Save merge list : Saves the log of merged nodes to a file which you can give a distinctive name. Nodes from any NodeClass which were merged are saved to this list.

NOTE : Information on both deleted nodes and merged nodes is kept in a log file which ORA used to create delete and merge lists.

Clean : finds all nodes whose IDs are within the edit distance of the selected node. An edit is a change of a single character.
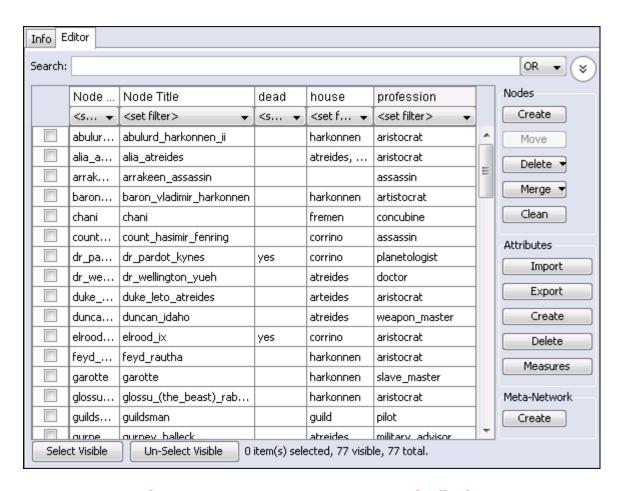
Find

Sub-set of a NodeSet

With the **Find** feature you can quickly locate specific nodes. Typing a term into the Search box will display all nodes with that term in their name or attributes.
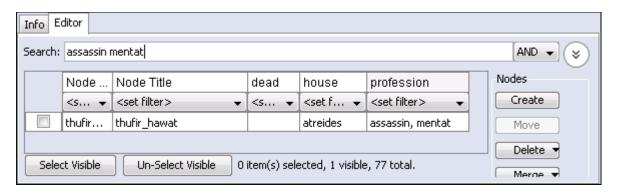
To the right of the Search box is the **AND/OR** selector. This is a boolean search tool. **AND** requires all terms in the Search box to be present in every node. **OR** will display all nodes with **any** of the Search terms present.

NOTE : Be aware of how **sub-strings** work in searches. A search for **male** will find all values of **male** as well as **female** because **male** is a sub-string of **male**.
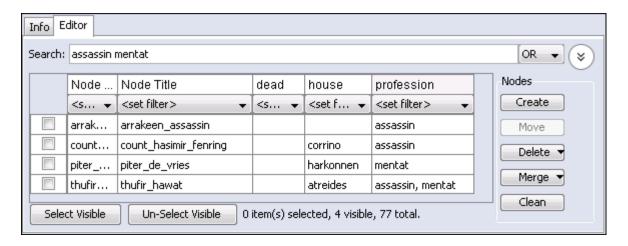
In the **Dune** dataset there is a profession attribute. Maybe characters have multiple professions.

Typing in **assassin mentat** brings up one entry for **thufir_hawat**. He is the only character with both professions searched for..
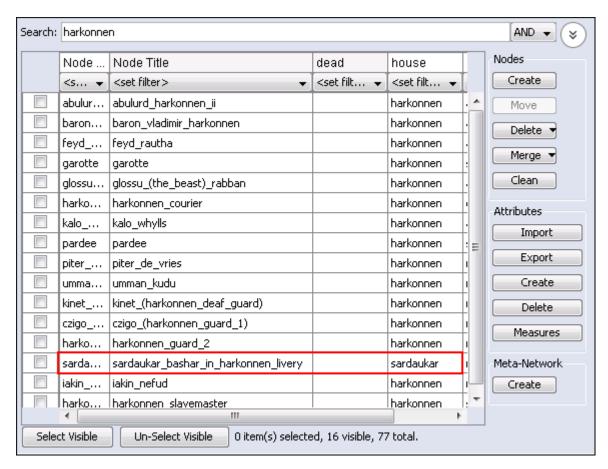
But to find all **assassins** and all **mentats** change the search parameters to **OR**. ORA now displays all characters with either profession.

Important note to understand about Search

The Search functions works on the nodeset as a whole meaning there's no selecting one single attribute to search. A term found anywhere marks that node to be displayed.

Below I searched for the term **harkonnen**. Naturally it displays all the characters that belong to the house Harkonnen. But notice there is one additional character displayed who belongs to a different house.
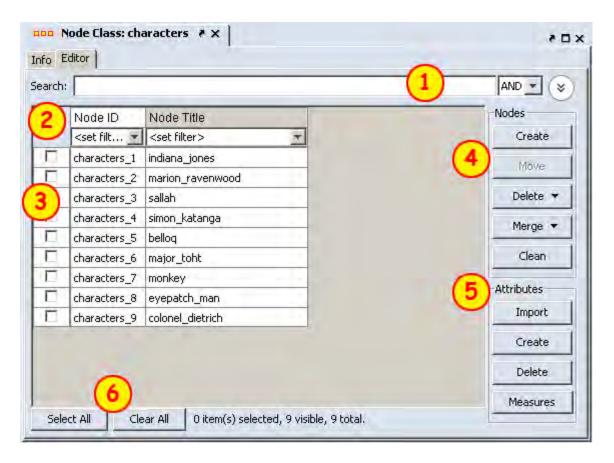
Info Tab - Nodeset

Panel 2 contains two tabs, **Info and Editor**. The Info tab is mainly designed to display information regarding the Meta-Networks, NodeSets, and Networks. But this is the area where you can rename the **Node Class ID** and the **Node Class Type**. Place the cursor in the box, make sure the entire name is highlighted, and type in your new name.

This area also gives you the **Node Count** of the selected Node Set as well as the **Attribute Names** contained within the NodeSet.

The **Editor tab** contains most of the editing functions.

- The **Search box** for finding particular node(s) in a set.

- The **<set filter>** for displaying only a particular sub-set of node(s).

- The **Check boxes** to designate which nodes to Delete or Merge.

- **The Nodes** buttons: Create, Delete, Merge, and Clean.

    The **Delete** button has these options.

- **Deleted selected nodes :** Deletes any nodes which have been check-marked.

- **Clear delete list :**

- **Load and apply delete list :**

- **Save delete list :**

    The **Merge** button has these options.

- **Merge selected nodes :**

- **Clear merge list :**

- **Load and apply merge list :**
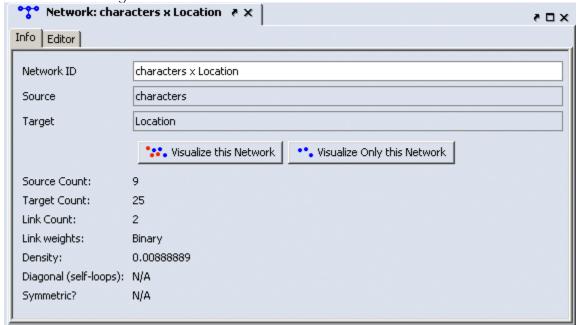
- **Save merge list :**

- **The Attributes** buttons: Create, Delete, and Import.

- Use the **Select All/Clear All** buttons to place, or remove, all check marks from section 3.
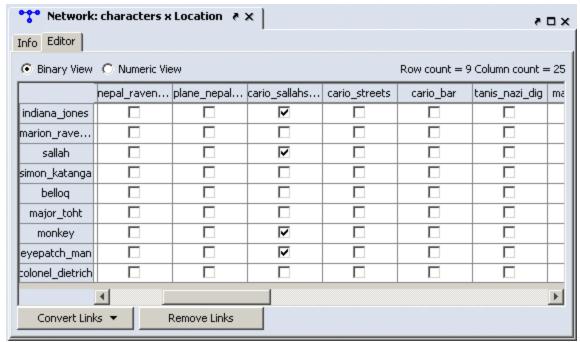
Info Tab - Network

- When a Network is selected there are also two tabs, **Info and Editor**.
- The Info tab contains a textbox, **Network ID** in which you can change the name of any network, The Source and Target Nodesets, and whether this network is directed or not, the Link Count, and the Density.
- The two buttons underneath allow the choice of **Visualize this Network** which will send the entire Meta-Network to the visualizer or **Visualize Only this Network** which display only the currently selected Network.
- Below is an image of InfoTabs and Editor Tabs in ORA.
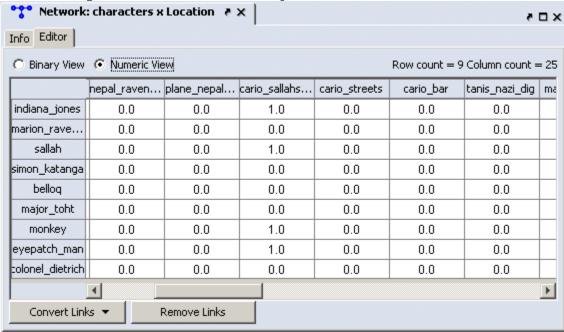
-
- The Editor tab has two views: **Binary and Numeric.**
- Binary view is used when you only need to establish a connection between two nodes with no weights. It's a simple **Yes or No** arrangement.

Network: characters x Location

Info Editor

⦿ Binary View   ◯ Numeric View   Row count = 9 Column count = 25

	nepal_raven...	plane_nepal...	cario_sallahs...	cario_streets	cario_bar	tanis_nazi_dig	ma
indiana_jones	☐	☐	☑	☐	☐	☐	
marion_rave...	☐	☐	☐	☐	☐	☐	
sallah	☐	☐	☑	☐	☐	☐	
simon_katanga	☐	☐	☐	☐	☐	☐	
belloq	☐	☐	☐	☐	☐	☐	
major_toht	☐	☐	☐	☐	☐	☐	
monkey	☐	☐	☑	☐	☐	☐	
eyepatch_man	☐	☐	☑	☐	☐	☐	
colonel_dietrich	☐	☐	☐	☐	☐	☐	

Convert Links ▾   Remove Links

-
- The Numeric view allows you to treat links between various nodes with more or less importance. Notice that all the **row nodes** have a check marked connection to the **column containing [yu]** in the binary mode. This denotes they know one another. But in the numeric mode this value is a 0.5 which is used to denote previous acknowledgement but is an adversary.

Network: characters x Location

Info Editor

◯ Binary View   ⦿ Numeric View   Row count = 9 Column count = 25

	nepal_raven...	plane_nepal...	cario_sallahs...	cario_streets	cario_bar	tanis_nazi_dig	ma
indiana_jones	0.0	0.0	1.0	0.0	0.0	0.0	
marion_rave...	0.0	0.0	0.0	0.0	0.0	0.0	
sallah	0.0	0.0	1.0	0.0	0.0	0.0	
simon_katanga	0.0	0.0	0.0	0.0	0.0	0.0	
belloq	0.0	0.0	0.0	0.0	0.0	0.0	
major_toht	0.0	0.0	0.0	0.0	0.0	0.0	
monkey	0.0	0.0	1.0	0.0	0.0	0.0	
eyepatch_man	0.0	0.0	1.0	0.0	0.0	0.0	
colonel_dietrich	0.0	0.0	0.0	0.0	0.0	0.0	

Convert Links ▾   Remove Links

-
- The **Convert Links** button contains options for converting links in a Network.
- **Binarize** link weights (x 0 -> x = 1)
- **Collapse** link weights (a=x=b -> x = 1

- **Negate** link weights (-x)
- **Invert** the link weights (1/x)
- **Logorithm** of the link weights (log10(x))
- **Absolute value** of the link weights ((|x|)
- **Row-Normalize** the link weights (x/OutDegree)
- **Increment** the link weights (c + x)
- The **Remove Links** buttons deletes all links for the selected condition and the value of the links you wanted removed.
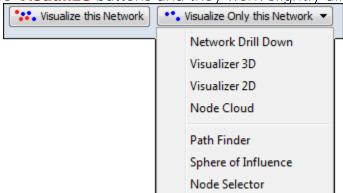
-
- **Less than**
- **Less than of equal to**
- **Equal to**
- **Not equal to**
- **Greater than or equal to**
- **Greater than**
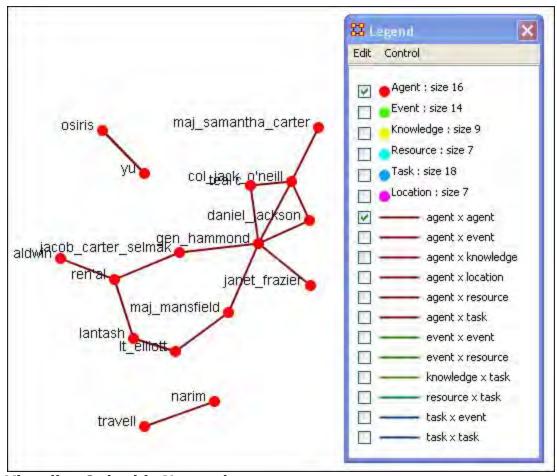
Visualize Buttons

- There are two **Visualize** buttons and they work slightly different.
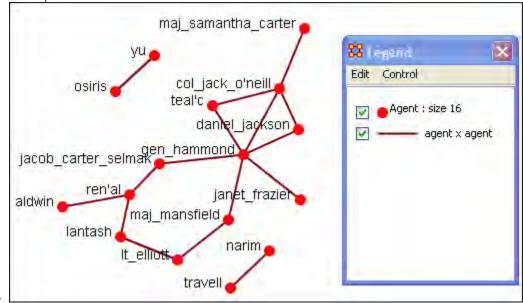
-
- **Visualize this Network**
- The **Visualize this Network** visualizes the chosen Network. In the **Legend** dialog box all NodeSets and LinkSets are available.

-
- **Visualize Only this Network**
- The **Visualize Only this Network** visualizes the chosen Network. But in the **Legend** only the NodeSet and LinkSet chosen are displayed. This is a much quicker means to display a Network. ORA only makes computations for the chosen Network.

-

- Clicking on the downward triangle brings up the dropdown menu with additional options for visualizations: **Network Drill Down**, Visualizer 3D, **Visualize 2D**, **Node Cloud**, **Path Finder**, **Sphere of Influence**, and **Node Selector**
-

File Menu

Below is an image of the File Menu accessible from ORA's main title bar menu. Further below are descriptions of the various tools accessible through the File Menu.

Open Meta-Network : opens a single Network with the following formats: DyNetML* (the native ORA xml format), .dl, .##h (UCINet), .csv* (comma separated values, Excel spreadsheet out), .txt (text files), .raw, .net (Pajek), .graphml*, .stl* (GMU model), and .lo (pathfinder).

Data Import Wizard : Contains additional options for importing Networks such as multiple file importing. See **Lessons** for more comprehensive information regarding this function.

Data Export... : Allows you to export a Meta-Network into a different format: **CSV, UCINET-Binary, UCINET-DL, Netdraw VNA, Text, Pajek, DyNetML, or C3Trace**.

Import CASOS Database...

Open Workspace... Allows you to open an entire workspace that you were previously using. A workspace consists of a set of Meta-Networks that were all open at the same time, and saved as a workspace.

CEMAP... Opens up the CEMAP Parser.

AutoMap... Opens up the program AutoMap.

Pile Sort... allows a user to load a series of entities such as concepts which can then be organized into a piles. The output will be a network in DyNetML format.

Save Meta-Network : This item is initially greyed out. It only becomes

active when you've made a change to a Network. You can use it as an indicator whether you've made a change to the active Meta-Network. Notice as you highlight each Meta-Network the icon in the toolbar will be greyed out (no changes) or colored (changes made to that Meta-Network).

Save Meta-Network As... : Allows you to save the changes in a Meta-Network under a new filename.

 Save Workspace : Saves the active Meta-Networks into a **Workspace** which can be opened up at a later date.

Exit : Closes the program and gives you two options for the next time you open up ORA.

- **Restore workspace :** If checked, the next time ORA is started it will load any Meta-Networks currently open when it was last closed.

- **Restore preferences :** If checked, all preferences from **Edit > Other...** will remain in effect. If not then ORA will resort back to it's defaults..

Open Files

ORA is designed to work with **DyNetML** files. However, data in a format other than DyNetML can be imported and converted.

Open Meta-Network

From the Main Menu select **File > Open Meta-Network.** to bring up the initial open dialog box. Navigate to the folder with your DyNetML file and click **[OK]**.

The selected file will then appear in **Panel 1** of the main interface.
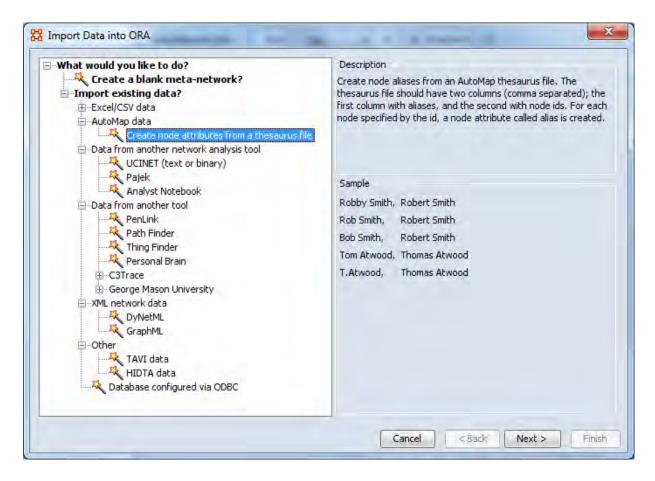
✨ Data Import Wizard

The **Data Import Wizard** imports files in a variety of File Formats. The idea behind the data import wizard is that users come to ORA with a variety of file formats from various different network analytical tools. Very rare is it that an analyst has a Meta-Network ready to load. Moreover, network data may exist in a separate file or in a variety of file formats, which then need to be merged together.

The Data Import Wizard is used to convert the user data from their native format and separate files and convert it into Meta-Networks and then into a DyNetML file.

The files can be in any of the following formats: **CSV, UCINET, GraphML, or Pajek**. ORA uses the file extension to determine the native file format to assign to your network. Therefore, if you change a file extension from its native extension **(e.g., GraphML file such as "example.gml" to "example.dot" ORA will not recognize the data format).**

This section will review the basics of the Data Import Wizard. See the lesson **Creating a Meta-Network** for more comprehensive information on importing.

Data Import Wizard is accessed either **File > Data Import Wizard** or the icon in the toolbar. You are presented with a dialog box showing all the file format options to import. Each contains a description on the right-side of the box.

Other Functions Besides Importing

- **Create a meta-network from separate network files:** Use this option to select multiple networks in one of the several native file formats indicated above and pull them into ORA. The file formats don't have to match. ORA will then combine all the network files into one Meta-Network, which will then be saved as a **DyNetML*** file (ORA's native file format).

- **Create a meta-network from table data:** Text files in the form of **rows of Nodes and columns of Attributes** will be used to populate existing NodeSets.

- **Import node attributes from table data:** Text files used to import Attributes to a Node.

Saving
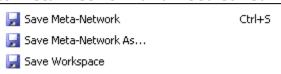
- Save Meta-Network

- This option is initially `grayed` out when you first load a Meta-Network. This signifies that the Meta-Network is currently unchanged. Whenever you make **ANY** change within he Meta-Network the option will become full-color indicating it is time save changes.
- After clicking the icon in the toolbar, or selecting the menu item, it will become grayed out again till you change your Meta-Network once again.

- **No change needs saved in the Meta-Network**

☐ Save Meta-Network	Ctrl+S
💾 Save Meta-Network As...	
💾 Save Workspace	

- **Changes made to Meta-Network and need saved**

💾 Save Meta-Network	Ctrl+S
💾 Save Meta-Network As...	
💾 Save Workspace	

- **Save Meta-Network As...**
- If you need to save a Meta-Network under a different name use select **File > Save Meta-Network As...**. This will leave the original file intact and create a new Meta-Network with the new file name.

- 💾 **Save Workspace**
- While working with multiple Meta-Networks you may want to retain the information of what files you had open for later use. Saving your workspace creates a file which can be opened later and will recreate your workspace as before.
- From the Main Menu select **File > Save Ora Project**. After selecting **[OK]** on the initial dialog box the **Save Ora Project** dialog box will appear. Navigate to the folder where you want to save your workspace.
- This Workspace can then be called up with the Load Workspace in a future session.
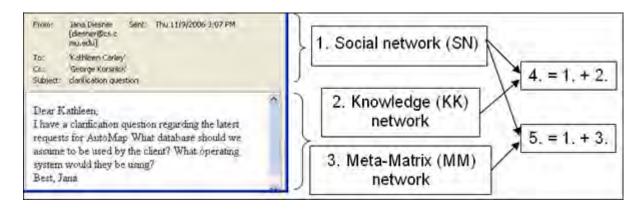
CASOS Email Parser

CEMAP is an ORA tool that primarily allows you to pull network data in the form of emails into ORA. However, the CEMAP tool could be used to convert any sources of data in the world into DyNetML, ORA's native file format, for analysis.

Setting up CEMAP is similar to setting up a new ISP mail client. What follows are procedures to configure this tool and how to use it.

Each email has an identifying number in a long number showing the date, for example: [**20060308**042751s1500bii66e]. The **20060308** is in the **year/month/day** format and says this email was sent on March 8, 2006.

The CASOS Email Parser **(CEMAP)** that is launched through AutoMap/ORA (http://www.casos.cs.cmu.edu/projects/automap/software.html) enables the extraction of different types of network information from emails **(e.g. who exchanges information, who provides what information, etc.)**. The following image shows what types of information can be extracted with CEMAP.

Network type1 **Social network (SN)** represents social network data that can be extracted from email headers **(e.g. agent-agent networks)**. In Social Networks, entities represent people, and edges represent exchanged emails **(frequency count)**. This network type does not require any text coding in AutoMap.

Networks types 2 and 3 are extracted by performing text coding in AutoMap. More specifically, **Knowledge Networks (KK)** represent semantic network or mental models that are contained in the bodies of individual emails. In **KK**, entities represent knowledge items, and edges represent the co-occurrence of terms in text. For **Meta-Matrix (MM)**, texts are coded in AutoMap according to a taxonomy or ontology **(e.g. meta-matrix, while are ontologies can be specified by the user)**. In **MM**, entities represent instances of categories **(e.g. agent, knowledge, resources)** of the ontology, and edges represent co-occurrences of terms in texts.

Networks types 4 and 5 result from the combination of **SN with KK and MM**, respectively. In type 4, entities represent people and knowledge, and edges represent emails and mental models. In type 5, nodes represent the categories of the taxonomy as specified by the user, and the edges represent the co-occurrence of the terms that represent instances of the taxonomy in the corpus. For the creation of type 4 and 5, the extraction of type 2 and 3, respectively, is mandatory.

CEMAP stores all network data as DyNetML files **(a derivate of XML)**. This data can be analyzed with any package that reads DyNetML. We recommend using the ORA software for this purpose **(http://www.casos.cs.cmu.edu/projects/ora/index.html)**.

The Top text fields

- **Input File -or- HTTP URL**

  Input File -or- HTTP URL **–** Input Data Set. You specify the email client, inbox type, username, and password as semicolon separated values. The location can be either file or server **(e.g. pop.gmail.com;INBOX;JohnDoe;JohnDoesPassword)**.

  NOTE : The following information is what is needed to access an email account.

- **Server –** The IP address or URL of the server.

- **Port –** The port on the server to connect to.

- **UserID –** ID of the User.

- **Password –** The User's Password for accessing the Mailbox.

- **Mailbox/Folder –** The ID of the mailbox to use.

  lip>**File Directories**

The Directory Buttons

- This section allows the user to dictate where the end files will be written to.

- **Select Output Directory for full document bodies –** Directory to save the body of the email text.

- **Select Output Directory for document bodies (new content Only) –** Directory to save body of the email. Discards all the repeat material that is quoted from previous emails.

- **Select Output Directory for document bodies (forward content Only) –** Directory to save Forwarding content.

- **Select Output Directory for SubjectLine Bodies –** Directory to save the Subject Lines of the email.

- **Select Output Filename for Header Network file –** Directory to write a single containing the Header.

Check boxes at the bottom

- **Anonymize Agents** – Anonymize Agents changes the agents name (only agents) to A# (# is a number) in DyNetML.

- **SSL (Secure Sockets Layer) (For IMAP and POP3 only)** – Creates a secure connection between a client and a server, over which any amount of data can be sent securely.

- **De-Dupe Messages** – Ignores identical emails.

  The file prefix makes it possible to put all the files in the same directory

The Start Button

The **Start Extraction** button begins the process.

Extraction of network type 1 data:

1. This network type is needed to perform social network analysis (SNA) based on header information only.

2. In CEMAP, **Analysis Type**, select the type of your mail box.

3. In CEMAP, **Input Data Set**, specify your email client, inbox type, username, and password as semicolon separated values, **e.g. pop.gmail.com;INBOX;JohnDoe;JohnDoesPassword**

4. In CEMAP, **Enter Destination Directory for Texts**, put in a directory of your choice, **e.g. C:\textInput. The email bodies will be stored in AutoMap format (.txt)** under this directory. Make sure that this directory does exits on your machine.

5. In CEMAP, **Enter Destination File for Network**, put in a directory AND filename of your choice. The filename extension needs to be .xml. **e.g. C:\output\AA.xml**.

6. You are now ready to extract network type 1. In order to do so, press the **Start Extraction** button. As a result, CEMAP will extract network data from the email headers **(e.g. agent by agent)** and store it as one XML file under the directory that is specified in the **Destination File for Network**.

7. You can pick up the output file and load it into the SNA package of your choice. We recommend using the **ORA package** (http://www.casos.cs.cmu.edu/projects/ora/download.php).

To extract networks of type 2, perform Semantic Network Analysis in AutoMap.

To extract networks of type 3, perform Meta-Matrix Text Analysis in AutoMap.

To extract networks of type 4, merge networks of type one and 2.

To extract networks of type 5, merge networks of type one and 3.

CEMAP is a feature in AutoMap that will import email files into AutoMap to analyze the text in the emails, such as who exchanges information, who provides what information, etc.

To use the feature:

- In the AutoMap menu bar, select **Tools > Open CASOS Email Parser (CEMAP)**.

- In the ORA menu bar, select **File > CEMAP Email-Importer...**.

  Once that is selected the **CASOS — Email Analyzer** window will appear. The Email Analyzer window is where you will type in your information of what type of network you would like to create while importing your emails.

  ## The different types of imports are

  - ○ Full email

  - ○ Header information only

  - ○ Forwarded/New emails

  - ○ Subject lines only

  ## Type Selection

  Select the type of Format your email is in, from the drop down menu.
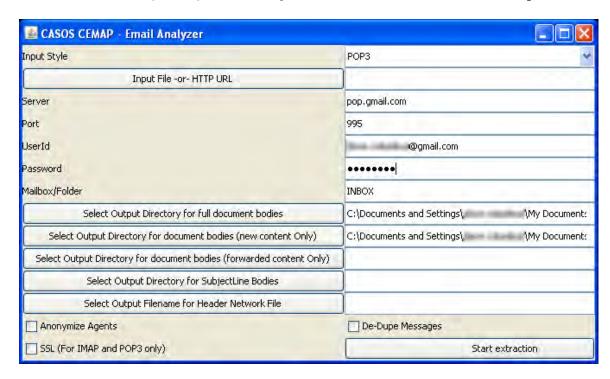
  - ○ **MBOX** – A generic term for a family of related file formats used for holding collections of electronic mail messages. All messages in an mbox mailbox are concatenated and stored as plain text in a single file. The beginning of each message is indicated by a line whose first five characters are "From" (the so-called "From_ line" or "'From' line") and a blank line is appended to the end of each message. For a while, the mbox

format was popular because text processing tools can be readily used on the plain text files used to store the e-mail messages. This is want you see when you **View Source** in a browser.
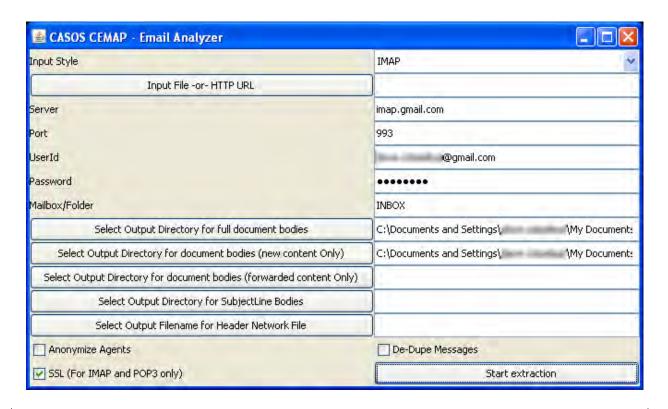
- o **POP3** – **(Post Office Protocol)** version 3 (POP3) is an application-layer Internet standard protocol to retrieve e-mail from a remote server over a TCP/IP connection.

- o **IMAP** – **(Internet Message Access Protocol)** Is an application layer Internet protocol operating on port 143 that allows a local client to access e-mail on a remote server.

- o **TXT** – Exported email files from outlook – will require a Data (file) set identifier

- o **CasosDB** – for internal use only

- o **DyNetml** – Proprietary format for ORA.

Once you have selected what type of format you would like to use, click on the **Start Extraction** button. This will begin the process of importing your emails.

Examples of POP3 and IMAP setups

Below are examples of setting up CEMAP with **POP or MAP** servers. Refer to the documentation with a particular server regarding the exact **Server, Port, or other pertinent information** necessary.
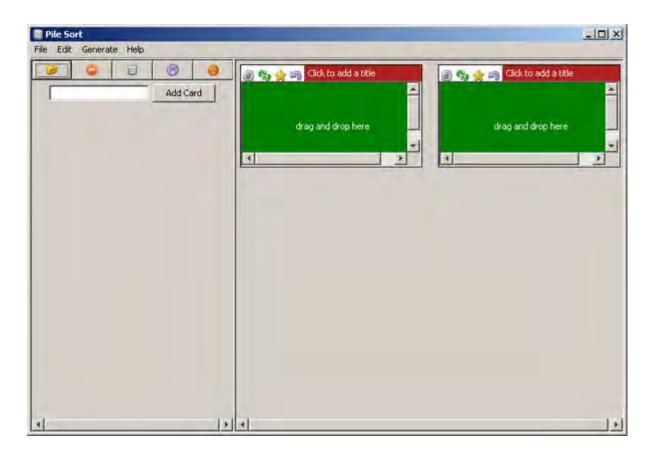
Pile Sort

The PileSort module allows a user to load a series of entities such as concepts or people's names. These entities are displayed in a deck, which can then be organized and grouped into a pile, which will contain entities that are similar to one another. The output of the PileSort module will be a network in DyNetML format.

The user can use either a text editor of a spreadsheet to create a two-column comma-separated file **(i.e. a .csv file)**. The first column contains the concept and must not be blank. The second column is **optional** and can contain metadata.

NOTE : Concepts can not contain embedded commas.

Pile Sort GUI

Pile Sort Interface : consists of two panes: The Side Deck and the Canvas. **The Side Deck** displays cards that have not been organized into piles, such as initially when the input file in read. **The Canvas** contains the piles where the cards are placed from the Side Deck.

Pile Sort Menus

File > Load a File :

File > Exit :

Edit > Add an Empty Pile : creates an additional empty pile on the canvas.

Generate > Create a Network DyNetML File : Creates a DyNetML file from the current state of organization. All cards in a pile are fully connected and cards remaining in the deck are isolates.

Help > Show Help Topics : Accesses the ORA Help files

Other Functions

Add Card : takes the concept that has been filled into the text field next to the **[Add]** button and creates a card with that name at the end of the card deck.

Transfer All Cards to Another Pile : Click and hold the icon of a pile and drag it to another pile. All cards in the first pile will be transfered to the new pile.

Cycle Through the Cards in the Pile : Rotates the cards in a pile that the card currently beneath the currently highlighted card becomes highlighted. The newly highlighted card becomes the pile's representative.

Return Cards to the Side Deck : Removes all cards from a pile and returns them to the side deck.

Mark a Card as a Pile's Representative : marks the currently highlighted card as a pile's representative.

Move Card to Another Pile : Click and hold a card in a pile and drag it to another pile.

Click to Add a Title : allows the entire pile to be given a name. Either click on the title bar or use the drop down menu item.

Drag and Drop Here : allows the user to drag cards from the side deck onto a pile.

Loading a list of concepts :

All piles are initially empty. The concepts appear in the side deck to the left. A scrollbar will appear if the number of concepts is greater than the screen length. An empty pile contains the banner **drag and drop here** signifying an empty pile. Once a concept is placed in a pile it will all concepts will be displayed in it's window.

NOTE : To have duplicate concepts in more than one pile requires the creation of duplicate cards.

NOTE : You can leave cards in side deck. Any card not so placed will become an isolate in the resulting network.
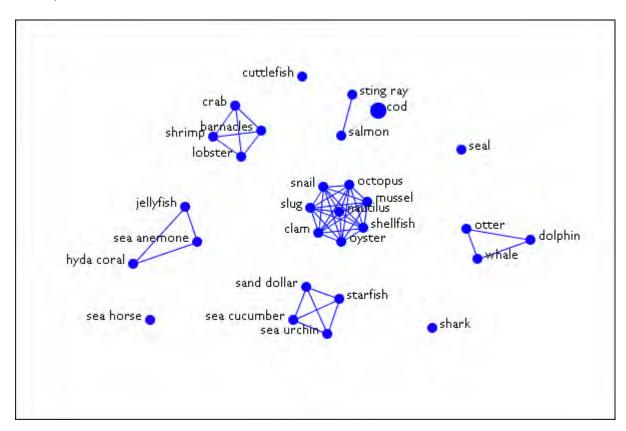
Viewing in ORA

Saving a sorted Pile : Sorted piles are saved to the DyNetML format. Networks are formed from the individual piles which include the metadata from each card.

Viewed in the Editor you can see all the saved information from Pile Sort. The column **containing_pile** is the name you gave the pile into which a card was placed. **meta-information** is the optional second column from the .CSV file originally loaded into Pile Sort. If that was empty, then the meta-information cell would also be empty.

Node ID	Node Title	containing_pile	freq...	meta_information	relative_freq...	seconds_on_t...
<set f... ▼	<set fi... ▼	<set filter> ▼	... ▼	<set filter> ▼	<set filter> ▼	<set filter> ▼
barnacles	barnacles	crustacean	I	crustacean	1.0	0.90
clam	clam	mollusk	I	mollusk bivalve	1.0	1.43
cod	cod	fish	I	fish	1.0	1.03
crab	crab	crustacean	I	crustacean	1.0	1.56
cuttlefish	cuttlefish		I	mollusk cephaloped	1.0	0.00
dolphin	dolphin	mammel	I	mammel	1.0	0.69
hyda coral	hyda coral	cnidaria	I	cnidaria	1.0	0.90
jellyfish	jellyfish	cnidaria	I	cnidaria	1.0	1.45
lobster	lobster	crustacean	I	crustacean	1.0	0.75
mussel	mussel	mollusk	I	mollusk bivalve	1.0	1.08
nautilus	nautilus	mollusk	I	mollusk cephaloped	1.0	0.95

Clicking the **[Visualize this Network]** button displays your Pile in the ORA Visualizer. How networks are formed is based upon the cards placed in each pile. The four cards not placed into a pile (cuttlefish, seal, sea horse, and shark) became isolates in the Visualizer.

 Edit Menu

The Copy/Paste functions are **dynamic** in nature. ORA understands what you have highlighted and will copy it onto the clipboard. Whatever is currently in the clipboard can be copied back into ORA any number of times.

The first two options are to Copy/Paste Editor values.

The copy function will place a copy of whatever is currently highlighted into the clipboard. That will remain until another item is copied.

The Paste function will create a copy of whatever is stored in the clipboard. This can be repeated as many times as necessary.

NOTE : ORA's menus change as different items are highlighted.. They also change after something has been copied. If a NodeClass had previously been copied the menu item will change to **Paste NodeClass**. If you then copy a Meta-Network the menu item will change to **Paste Meta-Network**.

Preference Menu

The View Menu shows a log of all activity performed in ORA.
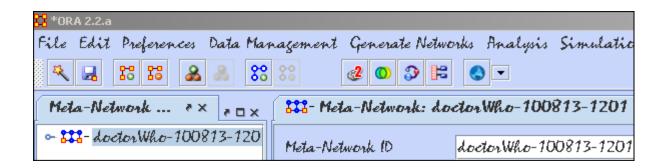
Defaut Layout : Resets the Editor GUI layout to its default panes position.

Change Font : Allows you to change the GUI font to any font on your system. Be careful with the font chosen as some fonts are built more for custom design rather than readability.

Load Font : Add font to use for the ORA display. ORA will ask whether to make this the default font. If **Yes** the the GUI will be displayed with the new font. These loaded fonts can be found in **C:\Documents and Settings\{username}\Ora Fonts**.

NOTE : Some fonts may make reading the GUI difficult. Take note of your previous font before making any changes.

Event Log : The log maintains a list of all actions performed in ORA. This allows you to review steps that were taken during a particular time.

NOTE : The report pane must be visible in the GUI to view the log. This pane can be reinstalled by selecting **View > Default Layout**.

Measures : Allows you to select the type of measures to run.
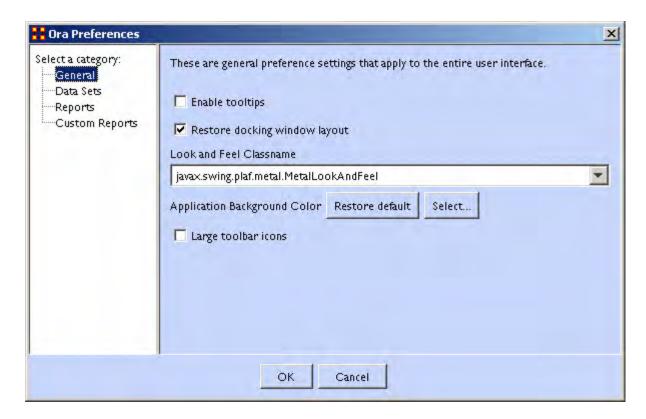
Use only fast measures :

Do not use slow measures :

Other : Contains settings for controlling the environment for working within ORA.

General
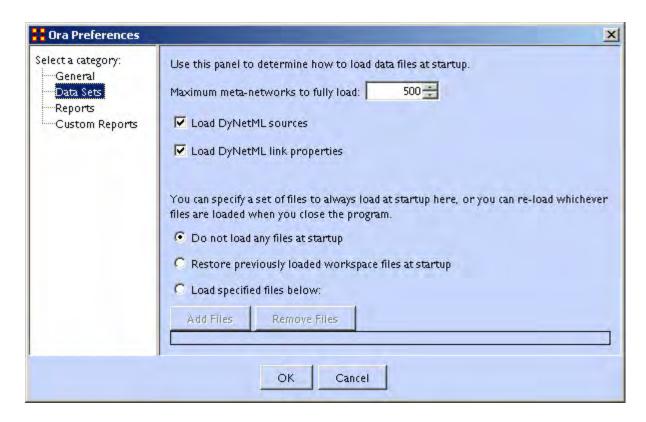
Enable tooltips toggles whether the tooltips are shown when you hover over an object.

Restore docking window layout will restore any changes you've made to the GUI layout. Unchecked at time of closing will result in ORA going back to it's default layout.

Look and Feel Classname

Large toolbar icons

Datasets

Maximum Meta-Networks to fully load

Load DyNetML sources
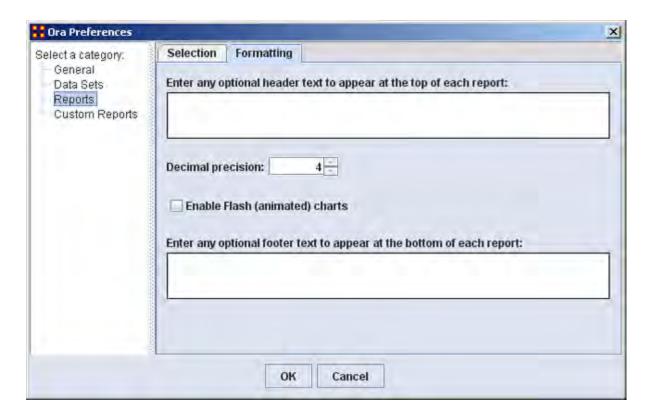
Load DyNetML link properties

Specify set of files to load at startup gives three options. **1) Do not load any files at startup** starts ORA with an empty Meta-Network pane. **2) Restore previously loaded workspace files at startup** loads the files stored in the Workspace file. **3) Load specified files below** allows you to pick which files you want loaded the next time you start ORA.

Reports
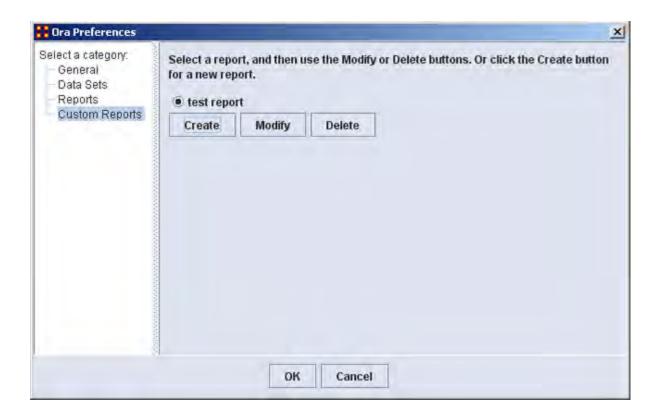
Select reports display as list of all reports available. Check marked reports will be used in the report generator. Report names can be changed by right-clicking a report and typing a new name. There is a **Reset report names** button which changes all names back to their default.

You can type in text for the **header and footer** of your reports. It will appear on all pages. The **Decimal precision** controls the number of decimal places for all numbers.
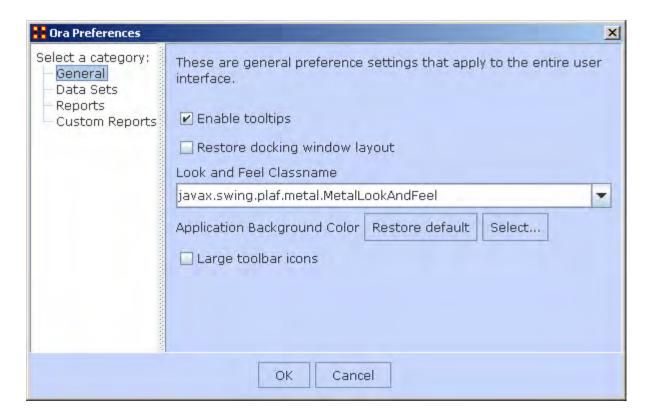
Custom Reports

Allows you to name, describe, and set up the measures to be used. If there are variable parameters for a measure they can be set also. If you have created any previous **Custom Reports** they will appear in this window. This is where they can be Modified or Deleted.

Preferences Panes

Preferences are global variations which control aspects of how ORA works.

General

Tooltips : Marking the checkbox shows tooltips when hovering over items. Remove the checkmark to disable tooltips.

Docking Layout : Marking the checkbox will retain the custom layout when ORA is restarted. No checkmark will fall back to the default layout.

Look and Feel : Sets the style of the ORA GUI.

Large Icons : Placing a checkmark toggles between using the 16x16 or 24x24 icons.

Data Sets

Maximum Data Files to Load : Tells ORA the maximum number of files to load upon start-up.

What To Load : You have three options for loading files when ORA is started up.

1. **Do not load any files at startup :** Ignores any files that were loaded during the last session.

2. **Restore previously loaded workspace at startup :** Loads the files that were current when ORA was last shutdown.

3. **Load specified files below:** Loads the files specified in the box below the option.

Reports

Selecting Reports to Display : Any reports with a checkmark will appear in the report generator window.

Changing/Reseting Report Names : Right-Clicking on a report name allows you to change the name. The **Reset report names** button resets all the names to their default

Import/Export Settings : If multiple people work on the same computer individual report settings can be saved for each user with **Export Settings**. Each user can use the **Import Settings** use their custom report names.

Report Headers : You can specify a header to print on the top of your reports.

Report Footers : You can specify a footer to print on the bottom of your reports.

Data Management

These sets of menu items enable you to control the addition or removal of Meta-Networks, Meta-Nodes, and Networks as well as adding Attributes to a Meta-Node.

Matrix Algebra, Meta-Network Transform, and Meta-Network Union… also create new Meta-Networks, Meta-Nodes, and Networks but do it with Mathematical operations.

Matrix Algebra… : Performs mathematical operations on two Meta-Networks.

Meta-Network Transform… : Cleans up a visualizer display making it easier to read.

Meta-Network Anonymize… – Resets the NodeID (and optionally the Node Title) to anonymous values.

118

Meta-Network Union – Creates a single Meta-Network from two (or more) existing Meta-Networks.

Meta-Network

Add New Meta-Network Used for adding or removing Meta-Networks from use.

Remove Selected Meta-Network : Removes any Meta-Network which is currently selected.

NodeClass

Add New Node Class... : Enables you to add, remove Node sets, as well as, change the size of the Node set. You can also add an Attribute to an existing Node Set. Depending on what you have selected different options may be greyed out.

Remove Selected Node Class : Removes all selected NodeClasses from the Meta-Network along with any associated Networks.

Add Attribute... : Creates a new attribute in a NodeClass. You need to give the attribute a name and also what type of attribute it is: **Text, Text Category, URI, Number Category, Number, Date**.

Network

Add Blank Network... : Similar to Node Class in that you can add or remove Networks. This function also enables you to **Set the Diagonal** of a graph to either **-1, 0 or +1**. Useful for setting consistent values on an **agent-by-agent** Network.

Remove Selected Network : Removes all selected Networks from the Meta-Network.

Set Diagonal : Allows user to constant values to the diagonal of a square network. If Binary View is chosen you can pick **True (+1), False (0), or True (-1)**. If Numeric View is chosen you can set the diagonal to any value.

Attribute Partition Tool : Creates new Meta-Networks from existing Meta-Networks using a node attribute.

Infer Links...

Beliefs :

Randomize Links : A new Meta-Network is created with identical NodeClasses, and with networks of the same size and density, but with links distributed randomly according to Erdos-Renyi.
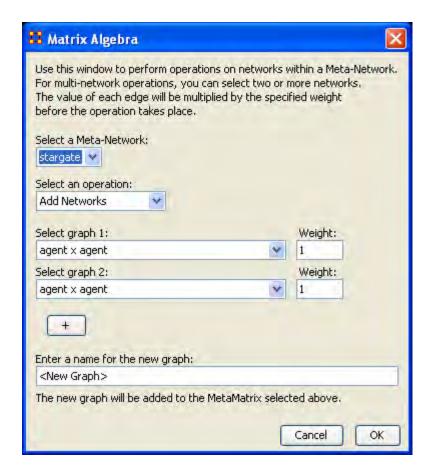
Network Algebra

Network Algebra allows you to perform mathematical operations on two Meta-Networks. The options in ORA are **addition, subtraction, and multiplication**. These functions help spot **strong and weak** connections within a Meta-Network.

NOTE : The term Network Algebra is synonymous with Matrix Algebra in computational parlance. Also, the following images display **Matrix Algebra** in the title bar. This may or may not be the case for the most recent version of ORA.

I start with one Network loaded.

From the menu select **Data Management ⇒ Network Algebra...**

The Network Algebra dialog appears. There are six areas which you can manipulate. These are each described below the dialog box.
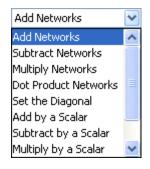
Note : Both Networks should be the same dimensions.

1. **Select a Meta-Network**

This dropdown menu will show all the Networks you currently have active. To perform a math function on a Meta-Network, highlight it in the dropdown menu.

2. **Select an Operation**

This option allows you to choose which operation, **Add, Subtract, or Multiply**, you want to perform on the Networks.

- o **Add:** Adds together corresponding cells of two different Networks.

- o **Subtract:** Subtracts corresponding cells of two different Networks.

- o **Multiply:** Multiplication is slightly different than the **add or subtract**. The number of rows in the first Meta-Network must equal the number of columns in the second. A Meta-Network can be multiplied by itself to help reveal specific **strong and weak** areas.

  - ▪ **Add Networks:**

  - ▪ **Subtract Networks:**

  - ▪ **Multiply Networks:**

  - ▪ **Dot Products Networks:**

  - ▪ **Set the Diagonal:**

  - ▪ **Add by a Scalar:**

  - ▪ **Subtract by a Scalar:**

  - ▪ **Multiply by a Scalar:**

  - ▪ **Divide by a Scalar:**

  - ▪ **Transpose:**

3. **Select Network 1:**

Next select the first Network you want to work with.

4. **Select Network 2:**

Then select the second Network to do the algebra on.

5. **Add Network:**

If you wish, you can add additional Networks to the equation. Different selections in the previous choices will give you different options in each dropdown.
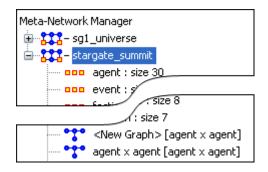
6. **Enter a name for the new Network:**
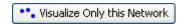
When everything is set, type in the name for your new Network.

Finally, select **[OK]** to create you new Network. You can then create more Networks if desired. When done, select **[Cancel]** to remove the dialog box from the screen.

Your new Network will now appear in the box with your Meta-Network. Next highlight the Network you just created.
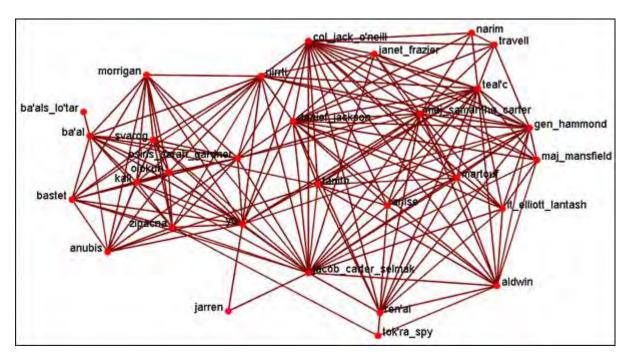
Next, select the **[Visualize Only this Network]** button
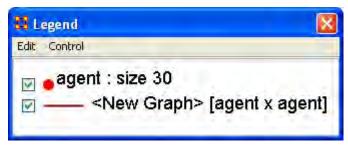
You will get a warning box asking if you're sure you want to visualize only one Network. Select **[OK]** to run the visualizer or close the box.
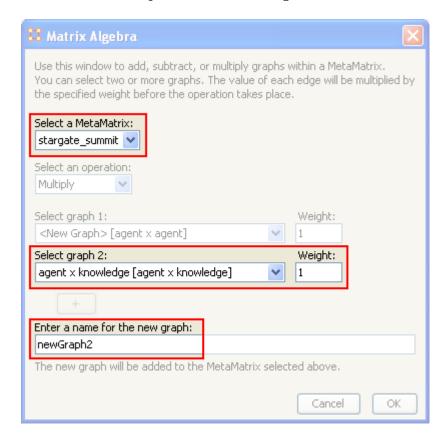
This will bring up the visualizer displaying the Network. Your display will depend on the Network selected. In this particular example there are only **Agents** and **Links connecting Agents** displayed in the Visualizer and the Legend.
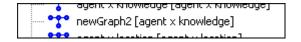
Now we'll create a second Network by using the multiply operation. The second Network will put knowledge into the mix.
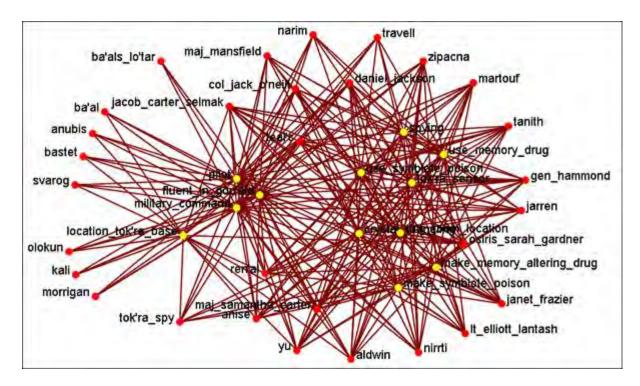
Note : Remember to name your file something different.
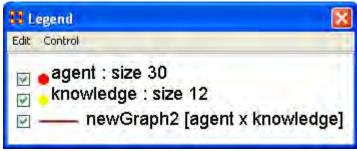
Highlight your newly created Network and select the **Visualize Only this Network** once again and select **OK** in the dialog box.

This time, since a Network with knowledge was selected the Visualizer and Legend display both **Agents and Knowledge** and also the **Links connecting them all**. Also the Matrix Algebra selected was multiplication it creates links which were not in the original Network.

Matrix Transform
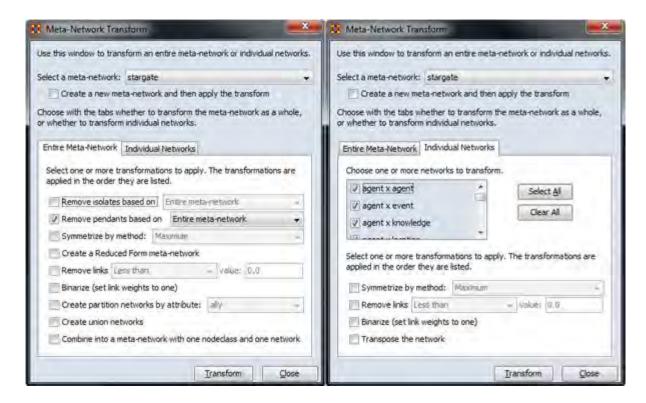
On some occasions the display in the Visualizer is too cluttered to make much sense of your network model. ORA offers a variety of ways to clean up the display in order to make it easier to view your data.

In the main menu select **Data Management > Meta-Network Transform…**.

The **Meta-Network Transform** dialog box will appear. It consists of three sections. The top section display a dropdown menu with all the currently active Meta-Networks. Use this menu to select the Meta-Network. The bottom section has two tabs. The **Entire Meta-Network** contains choices on how to transform the Meta-Networks. The **Individual Networks** tab allows you to select to work on **only** individual networks.

The "Entire Meta-Network" tab

This tab contains funtions which work on all nodeclasses and networks.

Remove isolates based on : allows you to remove isolates from the entire Meta-Network or from an individual Network.

Remove pendants based on : allows you to remove pendants from the entire Meta-Network or from an individual Network.

Symmetrize by method : turns directed or asymmetric network data into undirected or symmetric data. All three methods use a pair of values from the upper and lower portions of the table **(e.g. cell 1, 3 is compared to cell 3, 1)**. If a network had a link from A to B but not from B to A then symmetrizing the network would create a reciprocal link from B to A.

- **Maximum** Compares the values and uses the larger of the values. This is called **using the strongest value**.

127

- **Minimum** Similar to **Maximum** only it uses the lower, or weaker, value.

- **Average** takes the two values and averages them.

Create a Reduced Form meta-network : This is used to reduce the number of Networks in a Meta-Network when Link direction does not matter.

Remove Links : When using **weighted links** this functions allows for links to be removed from view based on link weight. You can choose from **six** options for the removal of links.

Binarize (set link weights to one) : Takes **all** links weighted greater than [**0**] and make them all equal to [**1**]. All links will be either [**1 or 0**].

Create partition Networks by attribute : This partitions the Node Set according to the values of the specified entity attribute and then creates a group of new graphs based on the partitioned Meta-Node. This kind of grouping is often done in network analysis to understand how some Meta-Nodes are **embedded** in various social roles or how they interact within larger Meta-Nodes. The dropdown menu will contain all the attributes in the meta-network.

Example, if you have an Agent Meta-Node that has a **gender** *attribute for each node that takes on values* **male** *and* **female***, then using this feature four new Agent x Agent Networks would be created that contained links from males to males, males to females, females to males, and females to females. No new Meta-Node is created and no data is changed.*

Create union matrices : Creates new networks that are the union of all networks of the same type. If the Meta-Network had two Agent x Agent Networks, then using this feature would create a new Agent x Agent Network which would contain all the links that were present in the original two Networks.

Combine into a meta-network with one nodeclass and one network : Creates a new Meta-Network with **a single** NodeClass and **a single** Network. The nodes are prefixed with the name of the NodeClass they were taken from.
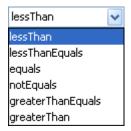
The "Individual Networks" tab

This tab allows you to **Choose one or more networks to transform** by placing a checkmark in the box. All marked netowrks will be processed.

Then select one or more transformations to apply :

Symmetrize by method : turns **directed or asymmetric** network data into **un-directed or symmetric** data. All three methods (**maximum, minimum, average**) use a pair of values from the upper and lower portions of the table (**e.g. cell 1, 3 is compared to cell 3, 1**)

Remove links : removes links based on one of six options.

lessThan	▾
lessThan	
lessThanEquals	
equals	
notEquals	
greaterThanEquals	
greaterThan	

Binarize (set link weights to one) : Take all weighted links greater than [**0**] and make them all equal to [**1**].

Transpose the network : writes the rows of network [**A**] as the columns of network [**A**^{T}] and writes the columns of network [**A**] as the rows of network [**A**^{T}]. It literally rotates the network 90° then mirrors the image horizontally.

	ceo	administrative_assistant	office_manager	accounts_payable	accounts_receivable
ceo	☐	☐	☐	☐	☐
administrative_assistant	☑	☐	☐	☐	☐
office_manager	☑	☐	☐	☐	☐
accounts_payable	☐	☐	☑	☐	☐
accounts_receivable	☐	☐	☑	☐	☐

	ceo	administrative_assistant	office_manager	accounts_payable	accounts_receivable
ceo	☐	☑	☑	☐	☐
administrative_assistant	☐	☐	☐	☐	☐
office_manager	☐	☐	☐	☑	☑
accounts_payable	☐	☐	☐	☐	☐
accounts_receivable	☐	☐	☐	☐	☐
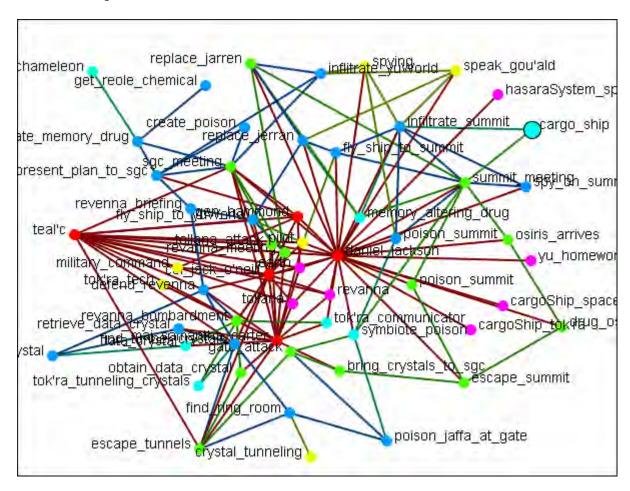
⊞ Meta-Network Anonymizer
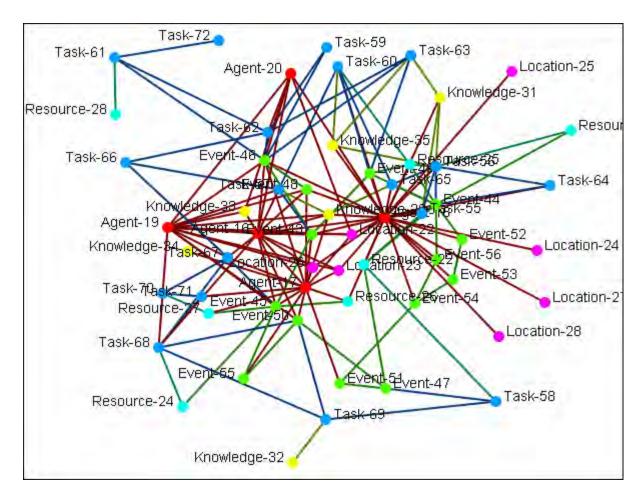
🔴 **Meta-Network Anonymize** tool removes IDs (and optionally Titles) from any chosen NodeClass. This is particularly helpful if you are discussing sensitive network information about real people and wish to remove their names from the network model.

The Visualizer

Visualizer Before Anonymizing : When you load your network model into the Visualizer, you will see (by default) the names of your agents or applicable entity class nodes. You may, for a number of confidentiality reasons, wish to hide this information from your audience yet still show them visually what the network can reveal.
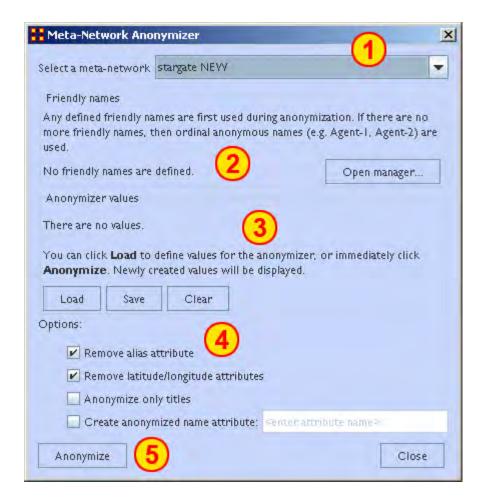
Visualizer After Anonymizing : None of the nodes have recognizable names and only the link and network information is known.

Meta-Network Anonymizer dialog

From the main menu select **Data Management > Meta-Network Anonymize…**.

The Meta-network Anonymizer tool will appear in a pop-up window. Here you can select a number of parameters pertinent to the node classes you wish to anonymize.

1. **Select a Meta-Network :** All active Meta-Networks appear in the drop-down menu. Select the Meta-Network to anonymize.

2. **Friendly names :** The user can create a list for each NodeClass of names that will not be anonymized.

3. **Anonymizer values :**

4. **Options :**

   o **Remove alias attribute :**

   o **Remove latitude/longitude attributes :** Removes geospatial attributes from the Meta-Network.

   o **Anonymize only titles :**

   o **Create anonymized name attribute :** If you would like to use descriptions other than Titles or IDS, you can generate your own terms. This is intended to be used with the friendly anonymizer tool but can be used for whatever reasons you deem fit. To do so, be sure the **create Name Attribute** is checked then enter your own term.

5. **Anonymize button :** Click the button when you are ready to crate an anonymized file.

Writing an Anonymized file : When the **[Anonymize]** button is selected ORA writes a **.CSV** file. The Save box allows the user to navigate to a directory to save the file.

```
Encoded ID,Original ID
A-1,A01
A-2,A02
A-3,A03
A-4,A04
```
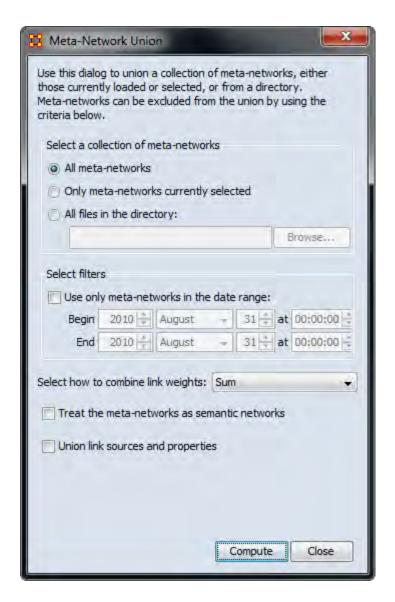
Node Title : If the **Anonymize node title** is selected then ORA will also write the original Title to the file.

```
Encoded ID,Original ID,Title
E-1,E01,revanna_meeting
E-2,E02,summit_meeting
E-3,E03,revanna_bombardment
E-4,E04,sgc_meeting
```

Meta-Network Union

Meta-Network Union is used to combine two or more Meta-Networks together. There are tools to select which Meta-Networks to use or exclude and how to combine the link weights. ORA can save out the result as an entirely new Meta-Network file.

Select a collection of meta-networks : allows the user to define the Meta-Networks to combine. It has two options.

1. **All meta-networks :** will create a new Meta-Network using all the currently loaded Meta-Networks.

2. **Only meta-networks currently selected :** will create a new Meta-Network using only the **selected** Meta-Networks. All nodes will be merged into one NodeClass and all links will be merged into one Network.

3. **All files in the directory :** will create a new Meta-Network from all the files in a specified directory. Use the [**Browse**] button to navigate to the directory to use.

Select filters : narrows the Meta-Networks to use. If the **Use only meta-networks in the date range:** is checked you will need to fill in the **Begin** and the **End** dates. This will filter out Meta-Networks which fall outside those dates.

Select how to combine edge weights: contains the options for how to reduce the amount of edges within a Meta-Network. The options are **Minimum, Maximum, Average, Binary, and Sum**.
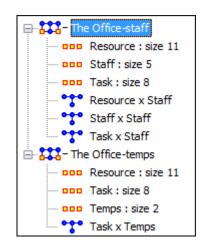
Treat the meta-networks as semantic networks :

Union link sources and properties :
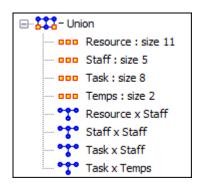
The Office-Union

I have two meta-networks for the office: **The Office-staff** and **The Office-temps**. Each has the same resource and task nodeclasses.

With only these two meta-networks loaded I choose **All meta-networks** and click the [**Compute**] button.

ORA combines these two meta-networks into one combining all the nodeclasses and networks.

Potential Problems Using Union

As great of a tool as this is there are a few things to look out for. First and foremost is Node ID conflicts. If the datasets contain radically different information but the same Node IDs you could end up with some results that might be confusing. We'll use the datasets from **Raiders of the Lost Ark** and **Star Wars**.

On the left are the characters in the raiders of the Lost Ark dataset. To the right are the Star Wars Characters. Notice the similarities and differences here. Raiders uses characters **character_1 through character_9**. Star Wars uses **Character_1 through Character_11** Notice the Upper-Lowercase C difference? Winder if that will present a problem..

Raiders   Star Wars

	Node ID	Node Title
	<set filt... ▼	<set filter> ▼
☐	characters_1	indiana_jones
☐	characters_2	marion_ravenwood
☐	characters_3	sallah
☐	characters_4	simon_katanga
☐	characters_5	belloq
☐	characters_6	major_toht
☐	characters_7	monkey
☐	characters_8	eyepatch_man
☐	characters_9	colonel_dietrich

Characters

	Node ID	Node Title
	<set filt... ▼	<set filter> ▼
☐	Characters_1	Princess Leia Organa
☐	Characters_2	R2-D2
☐	Characters_3	C-3PO
☐	Characters_4	Luke Skywalker
☐	Characters_5	Uncle Owen
☐	Characters_6	Aunt Beru
☐	Characters_7	Obi-Wan Kenobi
☐	Characters_8	Greedo
☐	Characters_9	Han Solo
☐	Characters...	Chewbacca
☐	Characters...	Darth Vader

Next, the locations. On the left are the locations in the raiders of the Lost Ark dataset **location_1 through location_19**. To the right are the Star Wars locations, **locations_1 through locations_13**. Another difference with Raiders using **location whereas Star Wars uses locations. That could be a potential problem also.**

Raiders   Star Wars

Node ID	Node Title	latitude	longitude
<set filt... ▾	<set filter> ▾	<set filt ▾	<set filt. ▾
Location_1	peru	-12	-77
Location_2	new_england	42.21	-71.05
Location_3	san_francisco	37.47	-122.26
Location_4	nepal-raven_saloon	28.05	84.40
Location_5	cairo-sallahs_house	30.02	31.22
Location_6	cairo-streets	30.02	31.22
Location_7	cairo-bar	30.02	31.22
Location_8	tanis-nazi_dig	30.97	31.09
Location_9	tanis-marions_tent	30.97	31.09
Location_10	berlin	52.30	13.25
Location_11	tanis-indys_dig	30.97	31.09
Location_12	tanis-airstrip	30.97	31.09
Location_13	desert_road	30.50	31.10
Location_14	cario-omars_square	30.02	31.22
Location_15	cairo-docks	30.02	31.22
Location_16	bantu_wind	33.30	28.38
Location_17	greek_isle-road	36.22	25.46
Location_18	greek_isle-tabernacle	36.22	25.46
Location_19	washington_dc	38.53	-77.02

Location

Node ID	Node Title
<set filt... ▾	<set filter> ▾
Locations_1	Tatooine
Locations_2	Mos Eisley Spaceport
Locations_3	Sandcrawler
Locations_4	Moisture Farm
Locations_5	"The Hills"
Locations_6	Rebel Blockade Runner
Locations_7	Death Star
Locations_8	Prison Block
Locations_9	Trash Compactor
Locations_10	Tractor Beam Engine
Locations_11	Control Area
Locations_12	Yavin IV
Locations_13	Millennium Falcon

So after a Union is run on these two meta-networks the 13 locations from Star Wars were combined with the 19 locations of raiders, each with having a separate entry. For this situation this is not a problem. But if your datasets contained information that needed to match up then you would end up with duplicate entries.

With the characters we find that the upper- lowercase [C] wasn't taken into account. There are still 11 characters. The problem that happened was it combined the information from the two datasets. We have Princess Leia Organa has a title of indiana_jones. That's definitely a problem.

Locations   Characters
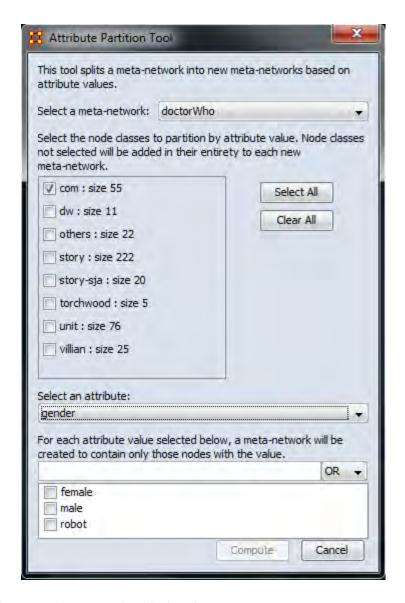
Node ID	Node Title	latitude	longitude
<set filt... ▾	<set filter> ▾	<set filt. ▾	<set fil.. ▾
Locations_1	Tatooine		
Locations_2	Mos Eisley Spacepor		
Locations_3	Sandcrawler		
Locations_4	Moisture Farm		
Locations_5	"The Hills"		
Locations_6	Rebel Blockade Runr		
Locations_7	Death Star		
Locations_8	Prison Block		
Locations_9	Trash Compactor		
Locations_10	Tractor Beam Engine		
Locations_11	Control Area		
Locations_12	Yavin IV		
Locations_13	Millennium Falcon		
Location_1	peru	-12	-77
Location_2	new_england	42.21	-71.05
Location_3	san_francisco	37.47	-122.26
Location_4	nepal-raven_saloon	28.05	84.40
Location_5	cairo-sallahs_house	30.02	31.22
Location_6	cairo-streets	30.02	31.22
Location_7	cairo-bar	30.02	31.22
Location_8	tanis-nazi_dig	30.97	31.09
Location_9	tanis-marions_tent	30.97	31.09
Location_10	berlin	52.30	13.25
Location_11	tanis-indys_dig	30.97	31.09
Location_12	tanis-airstrip	30.97	31.09
Location_13	desert_road	30.50	31.10
Location_14	cario-omars_square	30.02	31.22
Location_15	cairo-docks	30.02	31.22
Location_16	bantu_wind	33.30	28.38
Location_17	greek_isle-road	36.22	25.46
Location_18	greek_isle-tabernac	36.22	25.46
Location_19	washington_dc	38.53	-77.02

Node ID	Node Title	title
<set filt... ▾	<set filter> ▾	<set filt... ▾
Characters_1	Princess Leia Organa	indiana_jones
Characters_2	R2-D2	marion_rave…
Characters_3	C-3PO	sallah
Characters_4	Luke Skywalker	simon_katanga
Characters_5	Uncle Owen	belloq
Characters_6	Aunt Beru	major_toht
Characters_7	Obi-Wan Kenobi	monkey
Characters_8	Greedo	eyepatch_man
Characters_9	Han Solo	colonel_diet…
Characters…	Chewbacca	
Characters…	Darth Vader	

Attribute Partition Tool

To create new **Meta-Networks** which are a subset of nodes use the **Attribute Partition**. Select an Node-Class(es), select the Attribute set, then select the individual attribute(s) to form a new Meta-Network.

There are four sections to the dialog box:

Select a Meta-Network: The dropdown menu lists all the currently open Meta-Networks

Select Node Class(es): Place a checkmark in the box(es) of the NodeSets you want included.
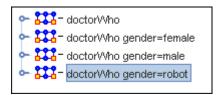
Select an attribute: This drop-down menu shows all the attribute(s) contained within the selected NodeSet(s). If you change the NodeSet(s) the choice of attribute(s) will be reflected in this dropdown.

Attribute Value: Place a checkmark in the box(es) of the attributes you want included in the new Meta-Networks. Each attribute check marked will create a separate Meta-Network.

139
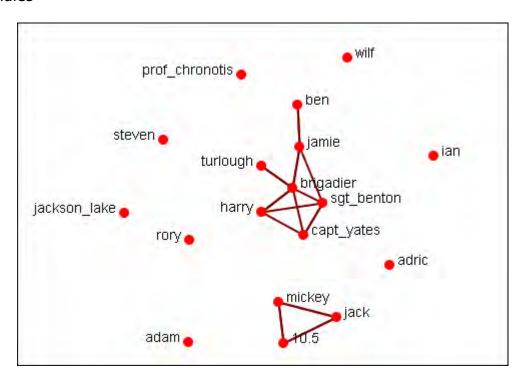
When you click the **[Compute]** button ORA will tell you how many new Meta-Networks were created and you will see them in pane 1 of the GUI.

Create New Meta-Network Example
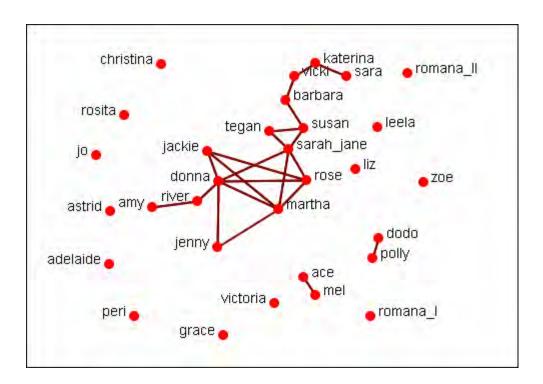
Using the Doctor Who Companion Network I selected the **gender** attribute. In the value section I placed check marks in all three boxes in order to create three new meta-networks. When I click the [**Compute**] button ORA creates three new meta-networks.
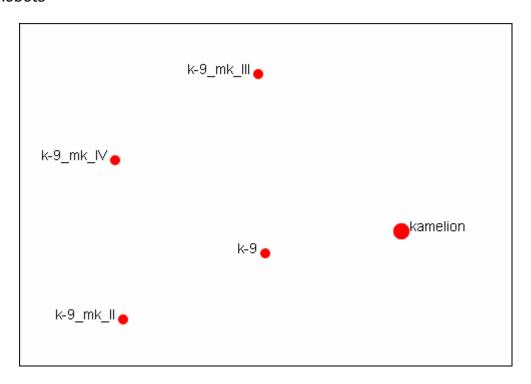
Males

Females

Robots

Randomize Existing Network

Randomizing Links is commonly done to test a hypothesis about a given network. If after conducting experiments, there is no significant difference between a hypothesis and a random graph, the evidence supporting the hypothesis could be weak.

A Random Graph : is obtained by starting with a set of **n nodes** and adding Links between them at random. Different random network models produce different probability distributions on networks.
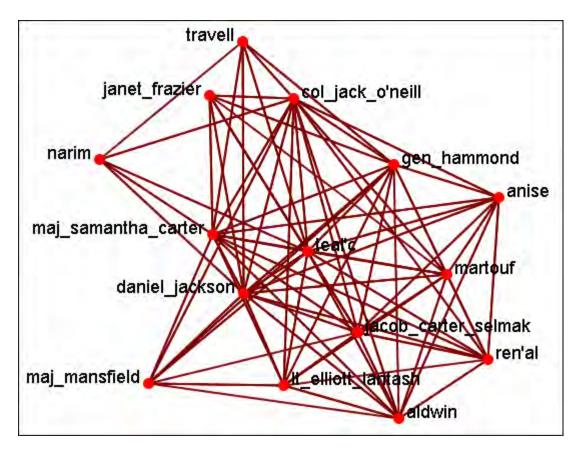
The theory of random networks studies typical properties of random networks, those that hold with high probability for networks drawn from a particular distribution. For example, we might ask for a given value of n and p what the probability is that G(n,p) is connected. In studying such questions, researchers often concentrate on the limit behavior of random networks-the values that various probabilities converge to as n grows very large. **Percolation theory** characterizes the connectedness of random networks, especially infinitely large ones.

Random networks are widely used in the probabilistic method, where one tries to prove the existence of networks with certain properties. The existence of a property on a random network implies, via the famous **Szemerédi regularity lemma**, the existence of that property on almost all networks.

Creating a Random Graph : is a function which **randomizes** the Links in a Network. From the main menu select **Data Management > Randomize Links**. From the dialog box select from the drop-down menu. A new Meta-Network will appear in Pane 1 and it's name will denote it has been randomized.
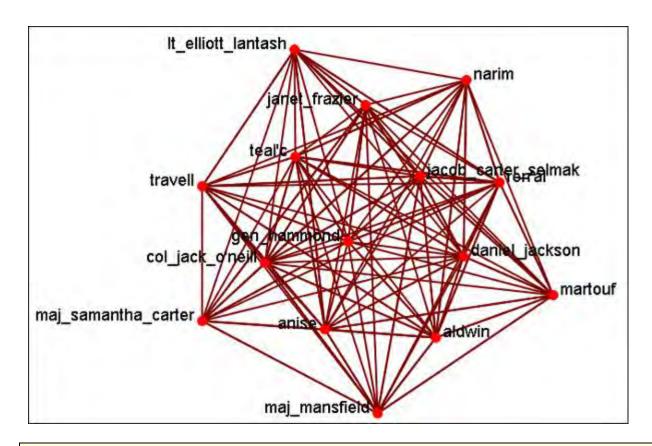
The original Meta-Network is selected and the Network **agent x agent** is chosen. From the **Info Tab** in the Editor the [Visualize Only this Graph] button is clicked. This displays the Network as it appears in the original Meta-Network.

Now select the newly created **randomized Network** and select the **agent x agent** Network. From the **Info Tab** in the Editor the
[Visualize Only this Graph] button is clicked. This displays the Network as it appears in the randomized Meta-Network.

As shown in the visualizer the links have been randomized.

Reduced Form Network

This is what Reduced Form does and does not do:

1. Uni-modal networks are not changed (**e.g. an AA does is NOT symmetrized**), but bi-modal networks are **symmetrized** in that the links from two or more networks are combined using the maximum method.

2. Bi-modal networks are combined: For example, if there is an AK and a KA network, then the KA is transposed to AK2 and then AK and AK2 are unioned with the Maximum link weight from each (a,k) or (k,a) being used.

3. Network names are normalized to have IDs agent x agent, agent x knowledge, etc.

11 FEB 11

Generate Network

144

Create New Meta-Network : Creates a new **blank** Meta-Network.

Influence Network Generator : Creates an Influence Network file for use in the external program Pythia.

Command and Control Structure Generator : Creates Network files for use in the external program Caesar III.

Create New Meta-Network In Visualizer : Creates a blank Meta-Network in the Visualizer which can be populated with NodeClasses and Networks from scratch.

Create Stylized Network : Generates new Networks using various Mathematical formulas including:

Erdős-Rény

Core-Periphery

Scale-Free

Cellular

Lattice

Small-World Networks

Create Expected Interaction Network : Predicts the expected interaction of a network given known tasks and other available network data. You can create:

Expertise Correlation

Similarity Correlation

Distinctiveness Correlation

Resemblance Correlation

Influence Network Generator

Influence Network : is a hypotheses regarding **tasks, performance, event happening, and related events**. This functions covers how to generate an influence network from a social network using ORA and examines the probability of event happenings based on the state and influence of supporting nodes.

The **Influence Network Generator tool** in ORA takes a Meta-Network and outputs an influence network that is readable by another software program, **Pythia**. Pythia, a software tool developed by George Mason University, has the ability to reason about an influence network's structure extracted by ORA. Therefore, ORA and Pythia can work in conjunction to perform an influence network analysis. Essentially, ORA generates the network and Pythia tells you something about it.

ORA generates influence nets for Pythia from Meta-Networks. ORA can also read in a Pythia influence net and assess its structure. After ORA generates an influence network, the user then has to work with Pythia and a SME (subject matter expert) to get the probability distributions to make sense. The key advantage is by inferring the influence network from the meta-network via ORA you save a lot of the start-up time in performing an influence network analysis.
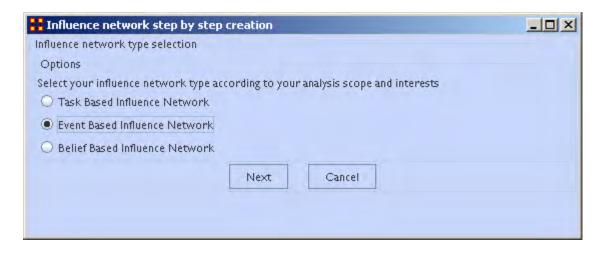
The ORA file generated by this function is directly loadable in Pythia. It is not a complete influence network and the user will need to make adjustments to the generated networks. It uses basic **ontological** interpretations and inferred baseline probabilities where the user specifies positive/negative probability.

We create an influence network based on: Extracting a task network which directs the target task to analyze. Then examine the complexity, importance, personnel/resource/expertise assignment of the tasks.
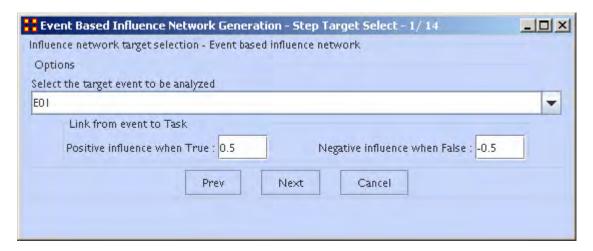
The examinations become influence network nodes containing ontological interpretation when: The task is done — the task is complex — the task is important — there are sufficient actors associated with the task — resources are available — expertise is available.

Then links are created as in the task network and background hypotheses nodes link to a task node.
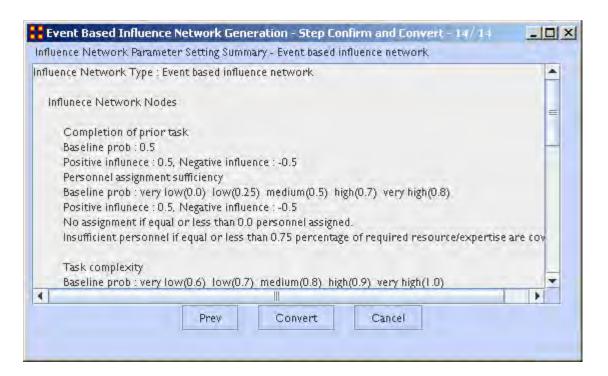
Running the Routine : From the main menu: **File > Influence Network Generator**. This brings up the **Influence network step by step creation** dialog box. There are three options: **Task Based, Event Based, and Belief Based**.

After selecting the creation type and clicking **[Next]** you will be presented with a number of steps depending on what information needs created.

Information on the thresholds you want and other data will be input in the dialog boxes until the final screen is reached.

Event Based Influence Network Generation - Step Confirm and Convert - 14/14

Influence Network Parameter Setting Summary - Event based influence network

Influence Network Type : Event based influence network

Influnece Network Nodes

Completion of prior task
Baseline prob : 0.5
Positive influnece : 0.5, Negative influence : -0.5
Personnel assignment sufficiency
Baseline prob : very low(0.0) low(0.25) medium(0.5) high(0.7) very high(0.8)
Positive influnece : 0.5, Negative influence : -0.5
No assignment if equal or less than 0.0 personnel assigned.
Insufficient personnel if equal or less than 0.75 percentage of required resource/expertise are cov

Task complexity
Baseline prob : very low(0.6) low(0.7) medium(0.8) high(0.9) very high(1.0)

[Prev]   [Convert]   [Cancel]

When all the information is input a dialog box with the results will display. This is the final check before clicking the **[Convert]** button and saving the file.

Il-Chul Moon, *Destabilization of Adversarial Organizations with Strategic Interventions*, PhD Thesis, School of Computer Science, Carnegie Mellon University

Extracting a Task Network for the Influence Network Generator
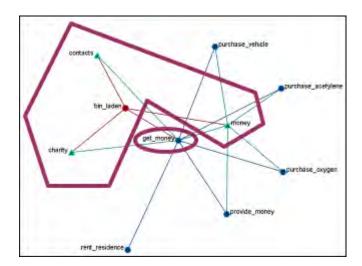
What follows are procedures to extract a Task Network for the Influence Network Generator.

In a Meta-Network there is an *event-to-event* network and a *task-to-task* network. Choose one task to analyze.

148
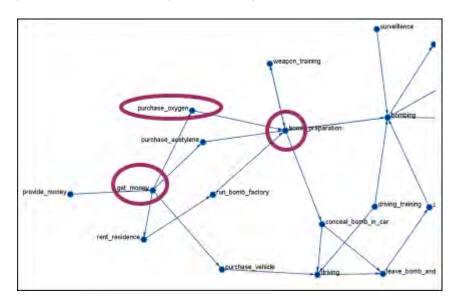
Here we look at **bomb preparation**. Find the tasks that need completed to accomplish the objective. Then trace from **provide_money**, **get_money**, **purchase_oxygen**.
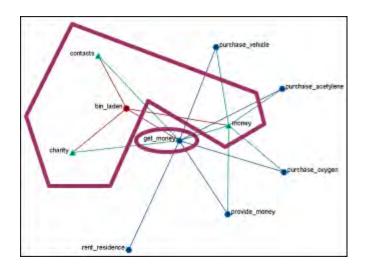
The complexity of the task depends on three items. **Resources / Expertise required and number of person required**.

- **Very Low (default baseline=1.0):** 0 resource/expertise required / 1 person required.

- **Low (default baseline=0.9):** 1 resource/expertise required / Less than 3 persons required.

- **Medium (default baseline=0.8):** Less than 4 resource/expertise required / Less than 4 persons required.

- **High (default baseline=0.7):** Less than 8 resource/expertise required / Less than 7 persons required.

- **Very High (default baseline=0.6):** More than 7 resource/expertise required / More than 6 persons required.
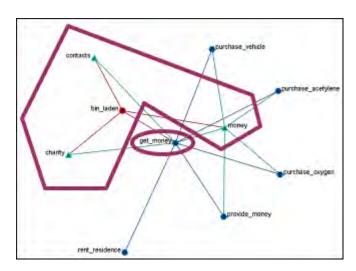
The importance of the task is determined by using **standardized degree centrality** and **betweenness** (ranging (0~1):

- **Very Low (default baseline=0.4):** Not connected to other tasks

- **Low (0.5):** Low out-degree density and low betweenness in task network

- **Medium (0.6):** Medium out-degree density and low betweenness in task network

- **High (0.7):** High out-degree density or medium betweenness in task network

- **Very High (0.8):** High out-degree density in task network

The actor assignment of the task is determined by sufficient actors, which are associated with the task.

- **No personnel (default baseline=0):** No assigned actors

- **Insufficient (0.25):** Only 75% of resources/expertise are covered by assigned actors.

- **Almost sufficient (0.5):** More than 75% of resources/expertise are covered by the assigned actors.

- **Sufficient (0.7):** At most one person associated with each resource and each expertise connected to the task.

- **More than sufficient (0.8):** More than one person associated with some resources and expertise connected to the task.

Resource / Expertise availability

Finds if **Expertise and Resource** are directly accessible. The availability is determined as follows:

- **Redundantly available/accessible baseline probability=0.75):** More than one person with the expertise associated with the task.

- **Available/accessible (0.6):** One person with the expertise/resource is associated with the task.

- **Unavailable/unaccessible (0.25):** No person with the expertise/resource is associated with the task.

Learn more...

Il-Chul Moon, Kathleen M. Carley, and Alexander H. Levis, <u>Inferring and Assessing Informal Organizational Structures from an Observed Dynamic Network of an Organization</u>, Management Science, 2008, Submitted and under review

Il-Chul Moon, Kathleen M. Carley and Alexander H. Levis, <u>Vulnerability Assessment on Adversarial Organization: Unifying Command and Control Structure Analysis and Social Network Analysis</u>, SIAM International Conference on Data Mining, Workshop on Link Analysis, Counter-terrorism and Security, Atlanta, Georgia, Apr. 26, 2008
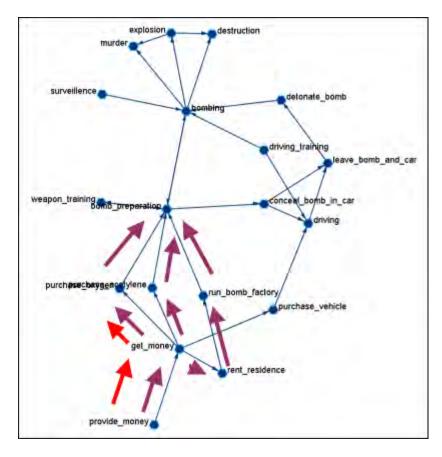
Pythia

Pythia is a timed influence net application. Conversely, data from Pythia can be loaded into ORA using the **Data Import Wizard**.
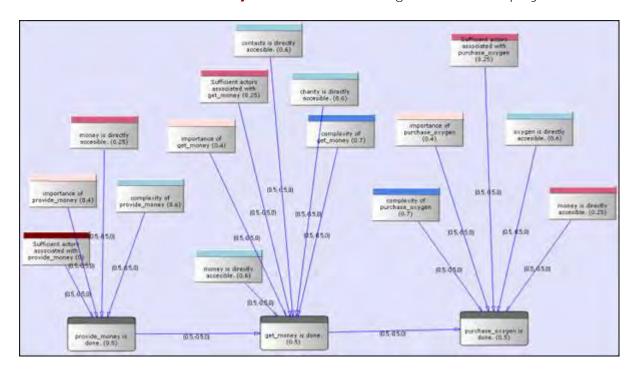
In conjunction with the Pythia application from George Mason University, data from ORA can be loaded into Pythia to create organizational displays.

What follows is a brief example of creating an organizational display using ORA with Pythia.

Start from an ORA **Task Network** (shown below)

The data can be loaded into **Pythia** to create an organizational display.

153

Reference at the George Mason University web site:
http://sysarch.gmu.edu/main/software/

Contact George Mason at the above URL for instructions on downloading Pythia and additional instructions on how to use the tool.

Learn more...

Il-Chul Moon, <u>Destabilization of Adversarial Organizations with Strategic Interventions</u>, PhD Thesis, School of Computer Science, Carnegie Mellon University

Command and Control Structure Generator

C2 Structure and Meta-Network

A Meta-network is a representation of a complex adaptive system. This system it has agents, expertise, resources and tasks. And inherent in this data are Meta-networks that contain the C2 structure of an organization, which are the links of a social network among commanders represented in the following ways:

- Command chains

- Information sharing relations

- Result/response sharing relations

However, meta-networks may only have one commander-to-commander network without classifying the nature of links. An analyst oftentimes needs a method to infer the nature of relations among commanders based on a given meta-network. That is where this tool can help. ORA uses the task assignment and communication directions to infer the nature of a link.

C2 structure in CAESAR III : Design View and Analysis View can be generated from a series of matrices describing the communications among decision makers (agents). The goal should be producing the matrices out of a social network. The analysis view has cognition aspects The nature of a link is determined by the timing of communication during the decision making process.

154

Meta-Network : Network among people (, knowledge, resources, tasks, etc). Data for statistical analysis, not exactly an analysis tool. Reveal the organizational structure, influence propagation, command and control structure, etc. Widely used in organizational science, counter-terrorism, structure reformation, etc.

C2 structure extraction in ORA : ORA provides a function, Command and Control Structure Generator. Automatic generation of C2 structure based on a target task and a social network among decision makers. Generated influence network is directly loadable in CAESAR III. It is not a complete C2 structure but a reasonable inference (or approximation).

C2 Structure Generation Procedure

Determine a target task to analyze. Extract the task network of the target task. Extract the related (assigned) agents of the task network. The other agents are aggregated and represented as the outside world (IN and OUT). Regard the agent-to-agent network as a information sharing network. Determine the network level based on the link directions and network topology. Then, extract a command chain from links bridging different network level. Determine the result sharing requirement based on the task network (i.e. the link direction between the two tasks indicate the result sharing requirement) Then, create a result sharing network based on the requirements.
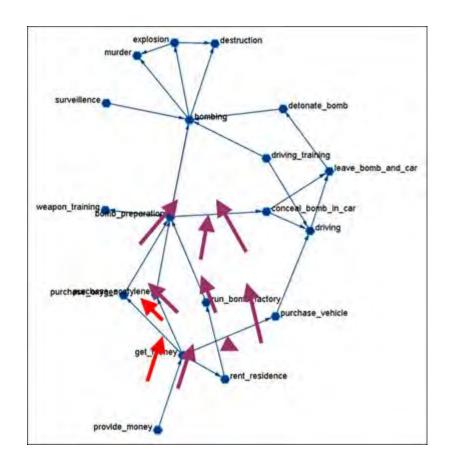
⊞ Extracting a Task Network for the Command and Control

What follows are procedures and an overview of extracting a *task network* for command and control analysis using ORA in conjunction with Caesar III, an external software tool for task network analysis.

In a Meta-Network, there are *event-to-event* networks and *task-to-task* networks contained in most data sets. We select one task to analyze and use ORA's command and control tool to discover the steps needed to complete this task from beginning to completion.
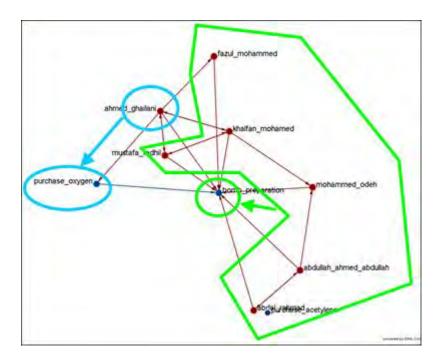
In the images below, we are extracting a task network needed for the bomb_preparation task node.

The idea is to find how tasks should be completed to accomplish the objective, bomb_preparation *(e.g., find the trace to a purchase_oxygen task. Provide_money —> get_money —> purchase_oxygen).*
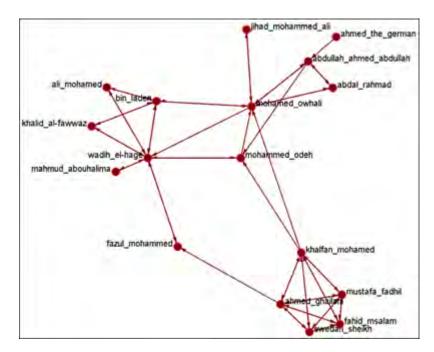
Assigned Agents

Identify the task network. Find all the assigned agents. Agents directly linked to a task in the task network.
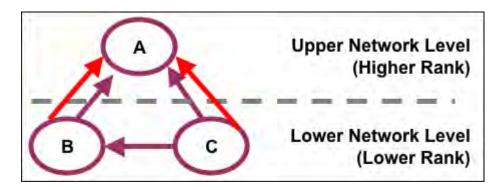
Approximated Information Sharing Network

Just the *agent-to-agent* network in a meta-network.
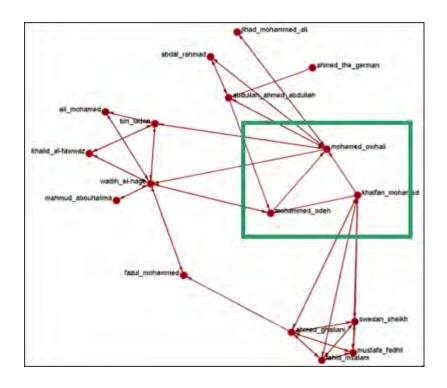
Inferred Command Chain

Determine the network level. Find the inferred hierarchy of the organization. Find the command chain from the hierarchy.
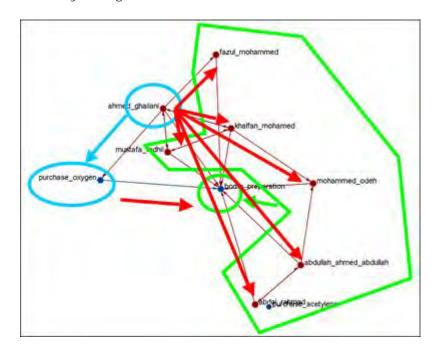
Red arrows are command chains. Purple arrows are social network links.

Inferred Result Sharing Network
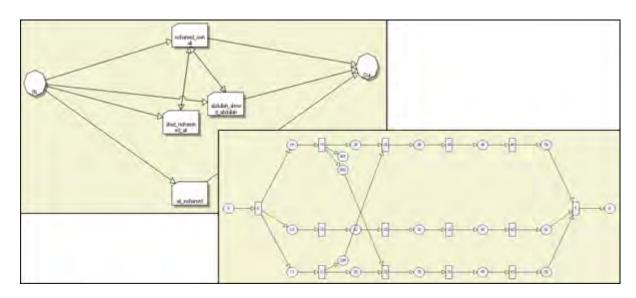
The task network defines the task prerequisites. Then, the result of a task should be informed to the agents doing the next task. Create a result sharing network by using the above inference rule.

Ahmed_ghallani should inform the result of purchase_oxygen to the agents doing bomb_preparation. Purchase_oxygen is a prerequisite of bomb_preparation.

Visualization by CAESAR III

Generated by ORA. Target task : surveillance. Design view and analysis view in CAESAR III.

Learn more...

Il-Chul Moon, Kathleen M. Carley, and Lee W. Wagenhals, Estimating Task Success Likelihood by an Influence Network Generated from an Organizational Structure, Organization Science, 2008, Submitted and under review
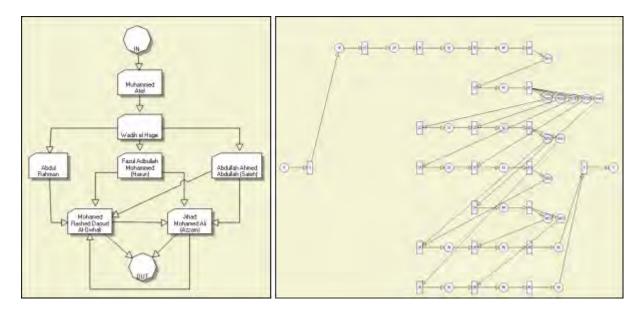
Il-Chul Moon, Eunice J. Kim and Kathleen M. Carley, Automated Influence Network Generation and the Node Parameter Sensitivity Analysis, International Command and Control Research Technology Symposium **(ICCRTS'08, DOD CCRP conference), Bellevue, WA, Jun 17**-19, 2008

 Caesar III

Caesar III is an software application for the design of decision making organizations at the operational and tactical levels; it takes into consideration cultural differences in coalitions and of adversaries. What follows in overview of how Caesar III and ORA can work together.

ORA can import data from Caesar III. See the Data Import Wizard for detailed instructions.

The **Design View and Analysis View** can be generated from a series of matrices describing the communications among decision makers (agents). The goal should be producing the matrices out of a social network. The analysis view has cognition aspects. The nature of a link is determined by the timing of communication during the decision making process.

Meta-Networks are generated by ORA. Then through **Caesar III**'s Design and Analysis Views their structure is analyzed.

> Reference at the George Mason University web site:
> **http://sysarch.gmu.edu/main/software/**

Contact George Mason University at the above URL for instruction on downloading Caesar III data.

Create Stylized Networks

Stylized Networks are models built upon common network structures evident in many real world environments.

Create Erdős-Rényi Network : A model for generating random graphs by setting an Link between each pair of nodes with equal probability, independently of other Links.

Create Core-Periphery Network : The core-periphery structure of the network has a dense central core of key agents and a large periphery of marginal agents.

The core-periphery structure is an ideal typical pattern that divides both the rows and the columns into two classes. One of the blocks on the main diagonal (the core) is a high-density block; the other block on the main diagonal (the periphery) is a low-density block. The core-periphery model is indifferent to the density of ties in the off-diagonal blocks.

When we apply the core-periphery model to actor-by-actor data (Core/Periphery), the model seeks to identify a set of actors who have high density of links among themselves (the core) by sharing many events in common, and another set of actors who have very low density of ties among themselves (the periphery) by having few events in common. Actors in the core are able to coordinate their actions, those in the periphery are not. As a consequence, actors in the core are at a structural advantage in exchange relations with actors in the periphery.

Create Scale-Free Network : Some nodes act as **highly connected hubs (high-degree)**, although most nodes are of low degree. Scale-free networks' structure and dynamics are independent of the system's size N, the number of nodes the system has. In other words, a network that is scale-free will have the same properties no matter what the number of its entities is.

Their most distinguishing characteristic is that their degree distribution follows a power law relationship, $\mathbf{P(k) \sim k^{-\gamma}}$,

$$\mathbf{P(k) \sim k^{-\gamma}},$$

Where the probability P(k) that a node in the network connects with k other nodes was roughly proportional to $k^{-\gamma}$, and this function gave a roughly good fit to their observed data. The coefficient γ may vary approximately from 2 to 3 for most real networks.

Create Cellular Network :

Create Lattice Network : A Lattice Network is a large-world network where nodes are placed at the integer coordinate points of the **n-dimensional** Euclidean space and each node connects to nodes which are exactly one unit away from it. **(i.e. if the lattice is two dimensional and the length of the lattice is 2 along the first and 2 along the second dimension, then it has 4 nodes and they're placed at coordinates (1,1), (1,2), (2,1), (2,2))** The two nodes are connected if the difference of one of their coordinates is one or minus one and all their other coordinates are exactly the same.

Create Small-World Network : A type of graph in which most nodes are **NOT** neighbors of one another, but most nodes can be reached from every other node by a small number of hops or steps. A small world network, where nodes represent people and links connect people that know each other, captures the small world phenomenon of strangers being linked by a mutual acquaintance.

The small-world behavior is characterized by the fact that the distance between any two inks is of the order of that for a random network and, at the same time, the concept of neighborhood is preserved.
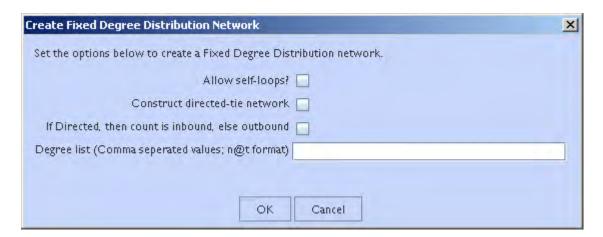
A Small-World Network is a regular network with a degree of random **long range** connections. It is rarely disrupted by the elimination of one node.

Fixed Degree-Distribution Network : A random graph is constructed by specifying as input: **(a)** the number of nodes in the network, and **(b)** the number of link for each node. The graph is random according to the input, however it should be noted that not all degree sequences are illogical so this algorithm may produce an empty graph as a result. **read more...**

Fixed Degree-Distribution Network

Create Fixed Degree Distribution : creates a random graph constructed by specifying as input: **(a)** the number of nodes in the network, and; **(b)** the number of links for each node. The graph is random according to the input, however it should be noted that not all degree sequences are illogical so this algorithm may produce an empty graph as a result.

Allow self-loops : Checked allows the generation of a node tied to itself. **Unchecked** disallows the generation of a node tied to itself

Construct directed-tie network : **Checked** generates directed network. **Unchecked** generates an undirected network

If directed, then count is inbound, else outbound : **Checked** the degree list is incoming ties. **Unchecked** the degree list is outgoing ties

Degree list : Enter the degree listing – in the desired node order. The input for the degree sequence is a **comma delimited list** of the number of edges. Correspondingly, the number of nodes is determined by the number of parameters entered.

To facilitate simplified entry, a shortcut can be made by specifying the number of nodes at a specific number of edges. Further, the node id in the generated graph will correspond with the position in the input string.

Degree sequences that are illogical will be impossible to generate, e.g., 2@20 *(two nodes with 20 ties each; impossible because the maximum ties a node can have in this network of size two is 1 tie.)* Further, some networks can be difficult to produce due to the specific degree distribution and the random process used to construct the network, so the algorithm is designed to make up to 10 separate attempts at constructing the exact distribution. After 10 attempts, the algorithm will return the error meta-network. The user may chose to resubmit the same request for another up-to-10 attempts at making the network.

For example, the following are valid input strings (with self-loops off and undirected):

- **1,5,3,0,1** - produces a 5 node network with one isolate and 2 pendants, etc.

- **3@2, 0, 0** - produces a 5 node network with 2 isolates and 3 nodes with 2 edges.

- **10@5,10@3** - will produce a 20 node network
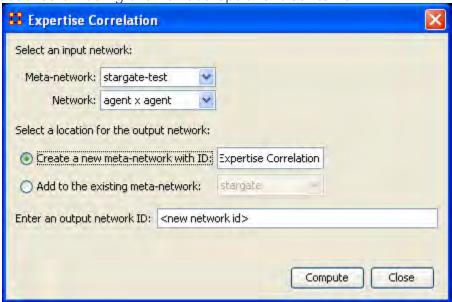
The following are valid input strings **but illogical**:

- **100,0,1** - is illogical and will produce a null meta-network named with an error message

- **3@2,1** - is illogical and will produce a null meta-network named with an error message

- **10@5,10@3,1** - is illogical will produce a null meta-network named with an error message

Create Expected Interaction Networks

- Below is an image of the Expertise Correlation tool in ORA and procedures on using the various options it contains:

-
- Below are links to the **Measures section** pertaining to each Correlation.
- **Expertise Correlation...** : Measures the complementarity of two agents based on their knowledge.

	col_jack_o'neill	maj_samant...	daniel_jackson	teal'c	gen_hammond
col_jack_o'neill	0.0	0.14285715	0.2857143	0.0	0.0
maj_samant...	0.14285715	0.0	0.2857143	0.14285715	0.14285715
daniel_jackson	0.16666667	0.16666667	0.0	0.16666667	0.16666667
teal'c	0.0	0.14285715	0.2857143	0.0	0.0
gen_hammond	0.125	0.25	0.375	0.125	0.0

-
- **Similarity Correlation**
- Measures the degree of similarity between agents based on the number of knowledge bits they both have.

	col_jack_o'neill	maj_samant...	daniel_jackson	teal'c	gen_hammond
col_jack_o'neill	1.0	0.33333334	0.25	1.0	0.5
maj_samant...	0.33333334	1.0	0.25	0.33333334	0.0
daniel_jackson	0.25	0.25	1.0	0.25	0.0
teal'c	1.0	0.33333334	0.25	1.0	0.5
gen_hammond	0.5	0.0	0.0	0.5	1.0

-
- **Distinctiveness Correlation...** : Measures how distinct are two agents based on the number of knowledge bits they hold oppositely.

	col_jack_o'neill	maj_samant...	daniel_jackson	teal'c	gen_hammond
col_jack_o'neill	0.0	0.22222222	0.33333334	0.0	0.11111111
maj_samant...	0.22222222	0.0	0.33333334	0.22222222	0.33333334
daniel_jackson	0.33333334	0.33333334	0.0	0.33333334	0.44444445
teal'c	0.0	0.22222222	0.33333334	0.0	0.11111111
gen_hammond	0.11111111	0.33333334	0.44444445	0.11111111	0.0

- **Resemblance Correlation...** : Measures the degree of resemblance between agents based on the number of knowledge bits they both have or both do not have.

	col_jack_o'neill	maj_samant...	daniel_jackson	teal'c	gen_hammond
col_jack_o'neill	1.0	0.7777778	0.6666667	1.0	0.8888889
maj_samant...	0.7777778	1.0	0.6666667	0.7777778	0.6666667
daniel_jackson	0.6666667	0.6666667	1.0	0.6666667	0.5555556
teal'c	1.0	0.7777778	0.6666667	1.0	0.8888889
gen_hammond	0.8888889	0.6666667	0.5555556	0.8888889	1.0

Analysis Menu

Generate Reports – is the top level for all the ORA reports. These are further grouped into similar reporting groups to make it easier to find the particular report you want.

Batch Mode Console

Measures Manager – The option allows you to choose which measures are to be used in generating reports for the selected Meta-Networks. The dropdown menu allows you to chose **measures lists** which divide them up into categories such as: **Slow Measures, Critical Risk Employees, and Resource Allocation Risk**.

Latent Semantic Analysis – Latent semantic analysis (LSA) is a technique in natural language processing, in particular in vectorial semantics, of analyzing relationships between a set of documents and the terms they contain by producing a set of concepts related to the documents and terms.

Correspondence Analysis :

Geary-C & Moran-I Analysis... : This function performs Moran's I test using phylogenetic and spatial link matrix (binary or general). It uses neighbouring weights so Moran's I and Geary's c randomization tests are
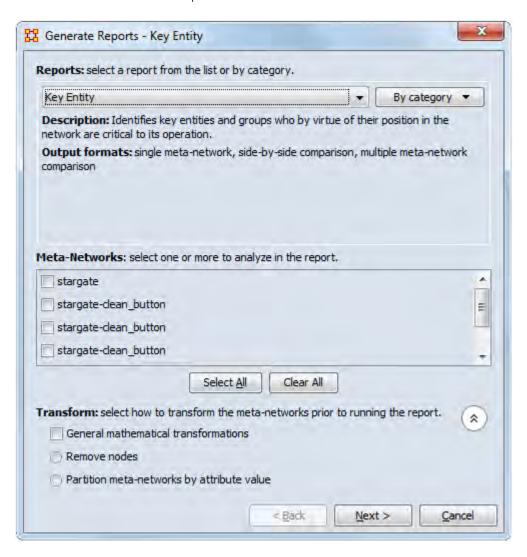
equivalent.

-

Generate Reports

Creates reports on the selected Meta-Networks. These reports are detailed in the **Reports Section**.

You can choose an individual report from the drop-down menu or use the **By Category** drop-down menu which forms them into groups by function. All measures are in both drop-downs.

After choosing a report from the drop down menu a description and output format will appear beneath it.

166

The next section allows you to chose which Meta-Networks to run the report on. Place a check mark next to all reports to use.

Clicking on the downward poiting double arrow reveals the **Transform section** which allows you to select the method the Meta-Networks are changed prior to running the report.

- **General mathematical transformation :**

- **Remove node :**

- **Partition nodes into separate Meta-Networks by attribute value :**

Click [**Next**]

This next screen will differ depending on which report was chosen. For some reports you will be taken directly to the print screen. For others there will be additional information you need to specify.

The Final Print Screen : allows you to chose whether to write the file to a **Text, HTML, CSV, or PowerPoint** file and tell ORA where to save it. You can also choose to use a different filename for all the reports written.

 Report Selection Wizard

 There are a lot of reports to choose from in ORA and you may have difficulty, at least initially, in determining which one is exactly right for you, especially if you are new to ORA. Likewise, the **Report Selection Wizard** helps you pick the right report by first prompting you with a series of questions relevant to the type of analysis you are seeking. Based on your answers, this tool will tell you the type of report to run and then it will run it. To access this tool in ORA, follow this path:

From the main menu select **Analysis > Report Selection Wizard**

Then you will see the following pop-up window (screen shot below):

Check the box(es) for the Meta-Network you need to analyze. Then click the **[Next]** button.

This will bring up a screen asking **Which of the following aspects do you want to focus on?**. This is the **Who, What, Where, or When** choice. Select one and click the **[Next]** button.

The next dialog box presents you with the question: **Please select one of the following:**. Choose one of the questions and click the **[Next]** button.

Do you want to know about the critical actors? The key word here is **critical**. This report will then find and rank the actors that are the most critical to your network, which can lend insight to **who**, if removed, might potentially impact the network the most.

Do you want to find a path between actors? The key word here is **path**. Path refers to the connections between any two actors in your network. For example, if Bob is connected Carol and Carol is connected to John, then Bob's path to John is 2 links (via Carol).

Do you want to know who someone influences or could possibly influence? The key word is **influence**. Typically one has influence, to some degree at least, on other people one connects to in a network. This is to say, if you have a connection to Joe and you are removed from the network, then Joe's connections to you are altered. You would be said to have a form of influence over Joe. However, some individuals may have influence that extends beyond a path length of 1.

Do you want to know if there is possible missing data? The key word her is **missing**. This report will help identify possibly segments of missing

data in your network if it appears certain conclusions are without proper network justification. Again, this is accomplished through algorithmic calculation. On a simple level, it might be said that if **Peter**, an accountant, has access to certain resources that only **Susan**, a director, has official access to; yet Peter uses those resources as well, it may be we are missing a link in our network data from Peter to Susan.

Do you want to know if there are groups? The key word here is **Groups.** Groups can sometimes be hard to spot visually, especially giving a complex and large Meta-Network. This report will reveal the groups present in the existing Meta-Network.

The group choices are as follows:

- Newman

- Concor

- FOG

Do you want to know what type of network this is? This question pertains to the topology of the network, or what sort of common network pattern your model could be classified into. This option will give you the **type**. For example**, cellular, lattice, small world**, etc.

ORA will display another information pop-up box, which tells you exactly the report ORA has determined to be most appropriate based on your data and your answers as to what information you are seeking to obtain about your network model.
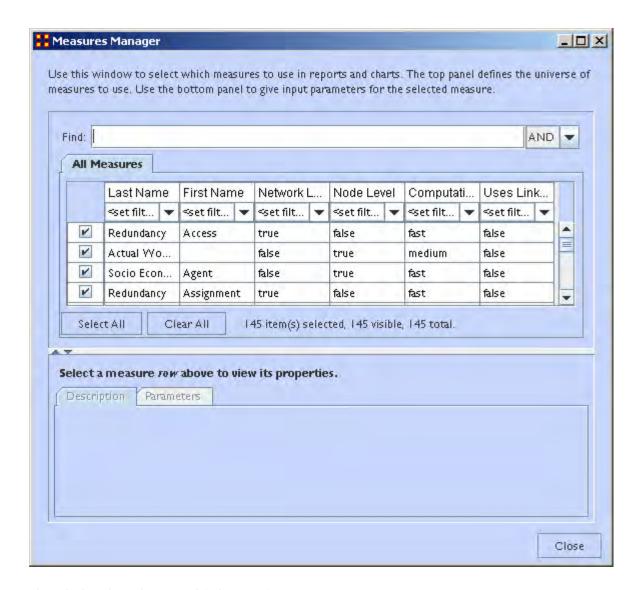
After selecting **[Finish]** ORA will generate it's reports and open them in both your browser and also in panel 3 of the main interface. Below is the beginning of the report run on the **Stargate** data set. Since this function has such a large number of possibilities, we will only show part of the beginning of the report in the interest of brevity.

Measures Manager

 Before you run any measures on a **Meta-Network** you can use the **Measures Manager** to set exactly the measures you wish to compute.

Form the main interface menu select **Analysis > Measures Manager.**

The dialog box has multiple sections:

Filter Commands :

This section comprises **two radio buttons** (Match at **least one** filter | Match **all** filters) which allow for complex filtering, a **[Reset Filters]** button, and a **Search text box** used for inputting your search. Whatever searches you input in this section are reflected in the **Tabbed Window** below. It will display only the nodes which match the search(es).

item(s) Selected | Visible | Total

When first called up the **All Measures** tab defaults to showing every measures available. When started there were **137 item(s) selected, 137 visible, 137 total**.

Selecting the **Slow Measures** tab this changes to **16 item(s) selected, 16 visible, 16 total**.

Typing **Cognitive** into the search text box reduces this to **16 item(s) selected, 8 visible, 16 total** showing that only 8 out of the 16 Slow Measures contain the word Cognitive.

Furthermore you can unselect measures by removing the checkmark next to the measure's name. With each measure deselected the number of **selected** items will reduce by one.

With these tools you can select only the measures you want to run reports on.

Buttons

- **Reorder**

- **Select All :** Selects (puts a checkmark in the box) all **visible** measures

- **Clear All :** De-Selects (removes check marks) from all **visible** measures.

Tabbed Window

This section contains tabs which sort the measures into various categories.

Each of the nine tabs sorts out the measures in different categories. This is very useful for large Meta-Networks when you'd prefer not run, say the **Slow Measures*** cause it would put too much a drain on your machine. By removing the check boxes for the Slow Measures they are taken out of the mix and the reports will effectively ignore them. You can turn them back on when you're finished by placing a checkmark back in the box.

In the **Tabbed Window** there are four columns:

1. **Last Name :** Deals the general type of measure **(Congruence, Exclusivity, Cognitive...)**. Most entries in this column will display one row as they are singular in what they need to show. Others will display two rows as it's nearly the same measure run on both **knowledge and resources**. Then a few **(like Centrality)** will display quite a few rows the types of measures written for that type of measures are somewhat varied.

2. **First Name :** Deals with the category of the measure **(Betweenness, Closeness, Centrality...)**. This is a secondary factoring.

3. **Graph Level :** **Graph Level*** produces a single number as output.

4. **Entity Level :** **Entity Level*** produces one number per entity.

Drop-Down Menu

Gives the user the ability to select which currently active Meta-Network to run the measures on, as well as, which individual networks.

Groupings of Reports

This chart shows the groupings for the reports in ORA

Groups and Which Reports They Contain

Groupings	Reports
Dynamics	Belief Propagation Change in Key Entities Immediate Impact QAP/MRQAP Analysis Statistical Change Detection
Geospatial	Geospatial Assessment Key Entity Trails
Groups	Core Network Local Patterns Locate Subgroups
Knowledge Networks & Network Text Analysis	Communicative Network Assessment Communicative Power Group Talk Hot Topics Large Scale Part of Speech Semantic Network
Locate Key Entities	Change in Key Entities Communicators Critical Sets Key Entity Management Simmelian Ties Analysis Sphere of Influence

	Standard Network Analysis
Locate Key Relations	**Shortest Path**
Specialty	**Capabilities** **Drill Down** **Influence Net** **Merchant Marine** **Optimizer** **Public Health** **Rules of Engagement** **Tactical Insight** **Trails Analysis**
Statistical Procedures and Diagnosics	**Context** **Missing Links** **Potential Errors** **QAP/MRQAP Analysis** **Statistical Change Detection** **Statistical Distribution**
Show Me Everything	

 # Reports In Which Group

ORA contains many reports for data analysis. The table below shows all the reports and into which groups they fall.

Reports and the Groups In Which They Belong

Report	Group(s)
Belief Propagation	**Dynamics**
Capabilities	**Specialty**
Change in Key Entities	**Dynamics** **Locate Key Entities**
Communicative Network Assessment	**Knowledge Networks & Network Text Analysis**
Communicative Power	**Knowledge Networks & Network Text Analysis**
Communicators	**Locate Key Entities**
Context	**Statistical Procedures and Diagnosics**

Core Network	Groups
Critical Sets	Locate Key Entities
Drill Down	Specialty
Geospatial Assessment	Geospatial
Group Talk	Knowledge Networks & Network Text Analysis
Hot Topics	Knowledge Networks & Network Text Analysis
Immediate Impact	Dynamics
Influence Net	Specialty
Key Entity	Geospatial Locate Key Entities
Large Scale	Knowledge Networks & Network Text Analysis
Local Patterns	Groups
Locate Subgroups	Groups
Management	Locate Key Entities
Merchant Marine	Specialty
Missing Links	Statistical Procedures and Diagnosics
Optimizer	Specialty
Part of Speech	Knowledge Networks & Network Text Analysis
Potential Errors	Statistical Procedures and Diagnosics
Public Health	Specialty
QAP/MRQAP Analysis	Statistical Procedures and Diagnosics Dynamics
Rules of Engagement	Specialty
Semantic Network	Knowledge Networks & Network Text Analysis
Shortest Path	Locate Key Relations
Simmelian Ties Analysis	Locate Key Entities
Sphere of Influence	Locate Key Entities

Standard Network Analysis	Locate Key Entities
Statistical Change Detection	Statistical Procedures and Diagnosics Dynamics
Statistical Distribution	Statistical Procedures and Diagnosics
Tactical Insight	Specialty
Trails	Geospatial
Trails Analysis	Specialty
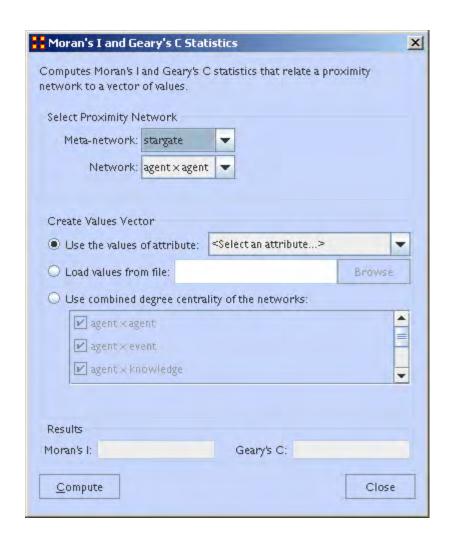
Moran-I and Geary-C

Description

Moran's-I : The measure of spatial autocorrelation to study stochastic phenomena which are distributed in space in two or more dimensions. The values of Moran's I range from **+1** meaning strong positive spatial autocorrelation, to **0** meaning a random pattern to **-1** indicating strong negative spatial autocorrelation. This particular statistic is designed for the measurement of spatial autocorrelation of ordinal, interval or ratio data.

Geary's-C : Based upon a paired comparison of juxtaposed map values and **ranges between 1 and 2**. Positive spatial autocorrelation is found with values ranging from 0 to 1 and negative spatial autocorrelation is found between 1 and 2. However, values can be found greater than 2 on occasion (Griffith, 1987). As similar values are in juxtaposition the numerator which measures the absolute difference squared between juxtaposed values will tend toward zero. Whereas, as non-similar values become juxtaposed the statistic will tend toward larger values in the numerator and thus toward its maximum value of two.

Select Proximity Network : Select a Meta-network from the first dropdown menu. The second dropdown menu will contain all the networks in the selected Meta-Network.

Create Values Vector : There are three ways to select a value

1. **Use the values of attributes :** uses an attribute within the Meta-Network.

2. **Load values from file :** browses for a file previously constructed.

3. **Use combined degree centrality of the networks :** Place a checkmark for one (or more) networks within the selected Meta-network to use. Combines the centrality to use in the formulas.

Results : Displays the **Moran-I and Geary-C** results.

Simulations

Near-Term Analysis : The **Near Term Analysis** (NTA) is a tool that allows for the removal of nodes from a given organizational structure to evaluate how the organization will likely perform as a result.

OrgAhead : OrgAhead is an organizational learning model designed to test different forms of organizations under a common task representation. Each member of the organization receives information from a subordinate or from the environment, makes a decision based on the information and what he or she has learned so far, and provides superiors with an answer to the decision. These decisions are either **yes/no** or **yes/maybe/no**; again, we're dealing with an intellective model.

Micro Simulations : Micro Simulation allows you to run some basic simulations on the curretly loaded visualization. Link weights are important as once they are normalized, they correspond to the probablity of a transmission.

Near Term Analysis

The **Near Term Analysis** (NTA) is a tool that allows for the removal of nodes from a given organizational structure to evaluate how the organization will likely perform as a result. The Near Term Analysis tool uses a multi-agent model as input, **Dynet ML***.

The goal is to provide an answer to the question on how an organization will behave and change after considering a sequence of strategic interventions or personnel loss by way of agent removal. In other words, we want to know what happens when we remove nodes from an organization, be they people, places, resources, knowledge sets or other common **Node Classes***.

This example will use the **Stargate dataset (SG-1)**. This dataset is included with the version of ORA downloaded and can be found on-line at the following url:
http://www.casos.cs.cmu.edu/projects/ora/sample.php.
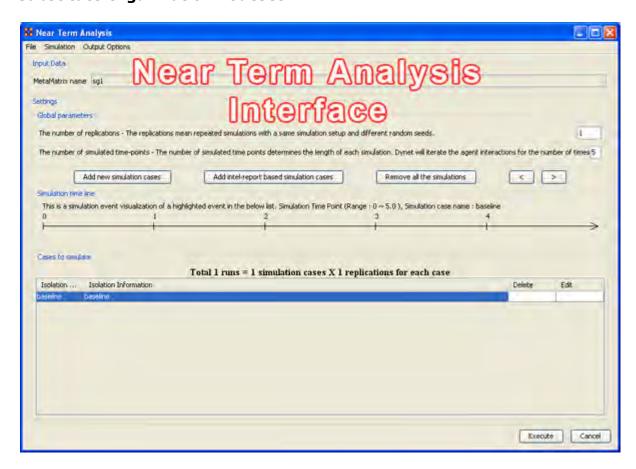
From the menu bar select **File ⇒ Open Meta-Network** then select SG1.xml.

In this example, we will remove two agents: **daniel_jackson** and **maj_samantha_carter**

1. Open the **SG1 Meta-Network** into ORA.

2. Highlight the SG1 Meta-Network **before** starting NTA in the ORA Meta-Network Manager panel.

3. From the main menu select **Analysis ⇒ Near-Term Analysis**.
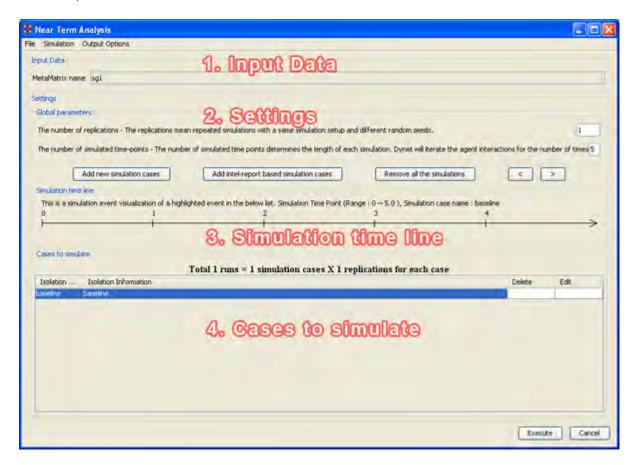
NOTE : If your dataset contains an Organization nodeset ORA will display the NTA main window. If not, it will ask for you to chose a substitute Organization nodeset.

When a Meta-Network is first loaded into ORA's the Near Term Analysis, a pop-up may appear asking for a **location** node. Select Cancel and proceed with the Near Term Analysis.

The NTA main window consisted of four sub panels: **Input Data, Settings, Simulation time line and Cases to simulate**. Below is a brief description of each panel:

1. Input Data

In this section is the file name of the Meta-Network loaded.

2. Settings (Global Parameters)

In the **Setting** area are two text boxes. The first, **The Number of replications** informs ORA how many times to run the simulation. The second, **The number of simulated time-points** defines the range in the section **Simulation time line**.

In the settings area, there are two critical factors: 1) the **number of replications** and 2) the **number of simulated time points**.

Why is this important?

The number of replications if set at a higher versus lower value, will result in more experiments being conducted on the scenario the tool is presented.

Therefore, the more replications the more likely the results will mirror those in a real world example.

The number of time points determines the length of the Near Term Analysis iterations. If this number is also set to a relatively high value, the simulation will take longer.

NOTE : Be mindful of any memory considerations when running high value replications or time points.

3. Simulation time line

This are of the Near Term Analysis tool shows a visual representation of the time points selected in Settings area above.
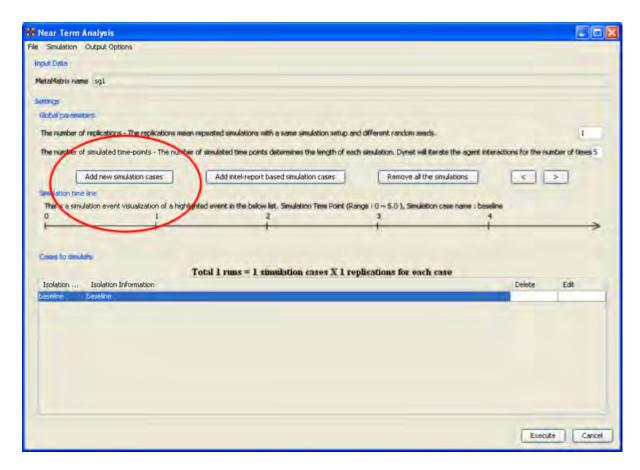
4. Cases to simulate

After the settings **(global parameters)** are set, the virtual experiment cells need set.
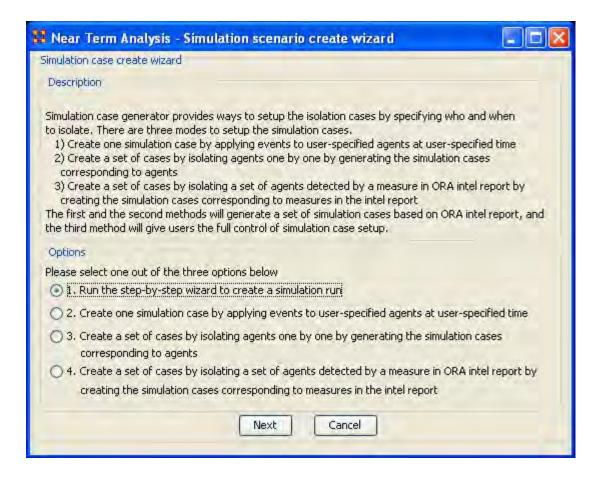
Initially, there is a pre-defined simulation cell, the **Baseline**. The Baseline is the cell without any entities removed as is the control situation when the organization does not experience any interventions. Besides Baseline, any other cells need set up. Start by selected **Add new simulation instances**, the leftmost of the four buttons.

Add new simulation cases

To run a Near Term Analysis, we will begin by selected the **Add new simulations cases** button from the main Near Term Analysis Tool (see area highlighted in red ellipse below).

After selecting **Add new simulation instances** button, a dialog box titled **Near Term Analysis– Simulation scenario create wizard** will appear. This wizard supports the creation of virtual experiment cells in three different ways. The **First** allows for the most freedom of choice. The **second** and the **third** options will ask for a set of criteria for selecting important agents in the network and make experiment cells according to the selection. Therefore, the first option gives full flexibility to users, and the second and the third option provides a systematic analysis setup method. To proceed this demonstration, select the first option and select **[Next]**.

There are four options:

The next section explores all four of the Near Term Analysis Options:

Run the Step-by-Step Wizard – This process will help walk through setting up a near term analysis experiment on the Meta-Network loaded into the Near Term Analysis tool.
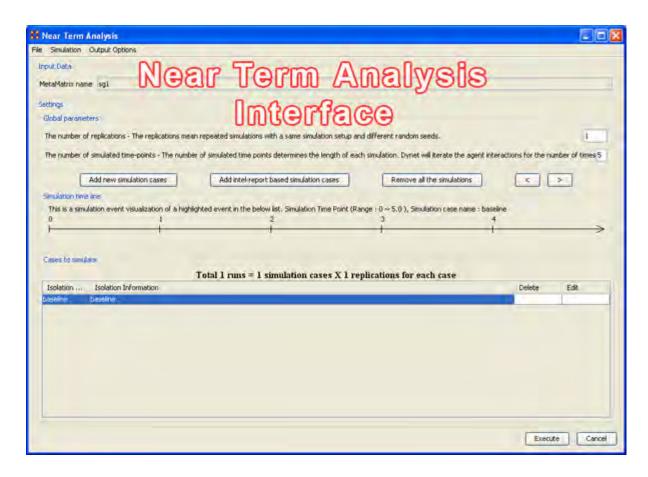
Create one simulation case – In this simulation experiment, events and user-specified time can be applied to agents.

Create a set of cases by isolating agents – This scenario isolates one agent and perform an experiment. Then the experiment can be performed again with the agent removed.

Create a set of cases by isolating a set of agents relating to a specific measure. – Based on measures contained in the ORA intel report, a set of agents can be isolated and a simulation case can be created.

 # Near Term Impact Menu

Save the current simulation setting: Brings up a save box where you can navigate to a directory and save **all** the current simulation settings.

Load a simulation setting from a file: Navigate to a directory to load a previously saved simulation setting.

NOTE : This will remove any settings that are currently active.

Close: Closes the Near Term Analysis window.

Execute simulation cases: Begins running the simulation. Same function as the **[Execute]** button at the bottom of the window.
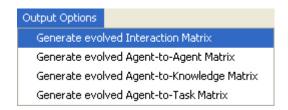
Show the add simulation wizard: Brings up the creation window. same as the **[Add new simulation cases]** button.

Add ORA-Intel report based simulation cases: Same as **[Add ORA-Intel report based simulation cases]** button.

Edit the highlighted case: Brings up the dialog box to edit parameters for selected event.

Remove the selected cases: Removes selected event from simulations.

Remove all of the cases: Same as the **[Remove all the simulations]** button.

These menu items will display in the **Near Term Results** menu:

Generate evolved Interaction Matrix:

Generate evolved Agent-to-Agent Matrix: Displays how agents interact with one another.

Generate evolved Agent-to-Knowledge Matrix: Displays how knowledge is diffused within the agent network
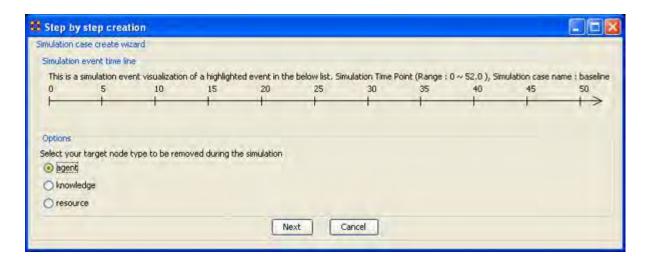
Generate evolved Agent-to-Task Matrix: Display how tasks are used within a network.
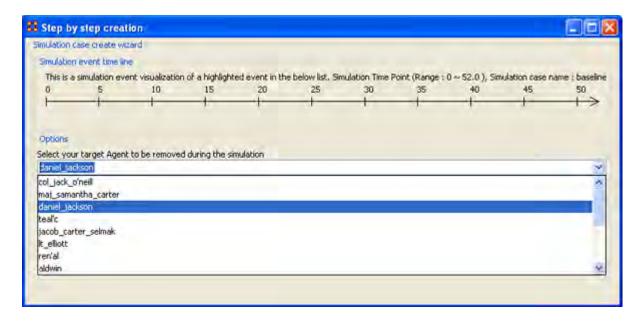
Near Term Impact Option 1

Step-by-Step wizard
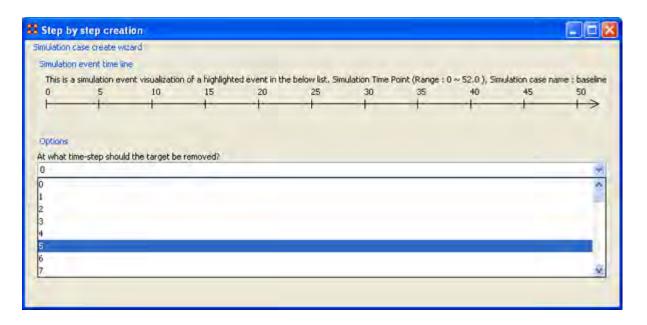
Selecting the first option brings up the **step-by-step creation** dialog box.
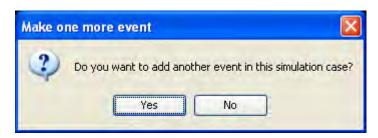
Under **Options** select whether to isolate an **agent, knowledge, or resource**. We'll leave it at agent. Then select the **[Next]** button.

Under **Options** select the agent to isolate. Here we select **daniel_jackson**. The select the **[Next]** button.
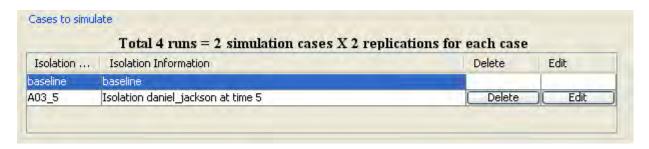
Under **Options** select the time-step to isolate the agent. For this simulation we select **5**. The select the **[Next]** button.
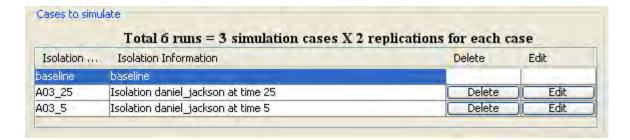
At this point you are asked if you want to add more events to the simulation or not.

You can add as many **events** as you wish. But for this example we'll only use only one event. Select **[No]** to continue.

Now repeat the steps above and add a **second** simulation. But this time let's remove **daniel_jackson** at time-step 25.

Now proceed to **Analysis Results**.

Near Term Impact Option 2
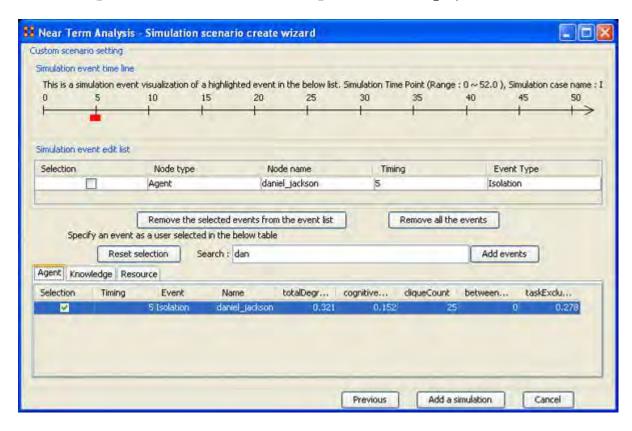
Apply events to specified agents

The second option for creating simulation files is to create one simulation case and apply events to specified agents, knowledge, or resource nodes.

Select the **[Add new simulation cases]** button to bring up the wizard.

1. Type in a search criteria to highlight only the agents to work with (i.e. typing in **dan** brought up only **daniel_jackson**.
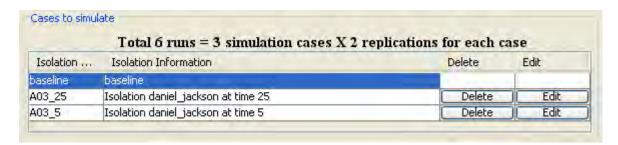
2. Place a checkmark on the line. Notice that when you placed a checkmark in the box the **Event** automatically changed to **Isolation**.

3. Double-click in the **Timing** cell to select it and type in a number from 0 to the highest number you selected for your simulation. In this case we'll repeat with the timing of "5".

4. Select the **[Add events]** button. This places the event in the **Simulation event edit list** section.

5. If there were more events to add you could continue with the same procedure. When you are done, select the **[Add a simulation]** button at the bottom.

The two buttons beneath the **Simulation event edit list** section to remove events from the list. The **[Remove all the events]** button obviously removes everything in the **Simulation event edit list** but the **Remove the selected events from the event list only removes the events with a checkmark beside them.**

The **[Reset selection]** button removes all check marks placed in the bottom table and sets the Event to **No Event**.

The **[Previous]** button returns to the main Wizard screen.

When you have finished you will have created the same two simulations as in Option 1.

Now proceed to **Analysis Results**.

 Term Impact Option 3
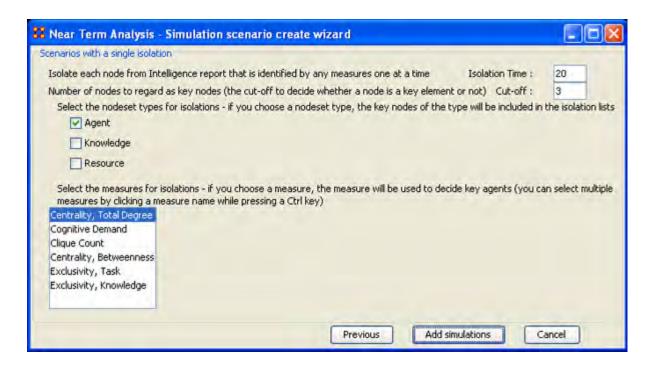
Create set of cases using Measures

Option 3 allows for selecting agents/knowledge/resources based on their ranking when certain measures are run on them.

Here are three examples run with the same **Isolation Time and Cut-offs**. Only the measures were changed.

Centrality, Total Degree

Isolation Case	Isolation Information	Delete	Edit
baseline	baseline		
daniel_jackson_20	Isolation null at time 20	Delete	Edit
jacob_carter_selmak_20	Isolation null at time 20	Delete	Edit
ren'al_20	Isolation null at time 20	Delete	Edit

Cognitive Demand

Isolation Case	Isolation Information	Delete	Edit
baseline	baseline		
ren'al_20	Isolation null at time 20	Delete	Edit
jacob_carter_selmak_20	Isolation null at time 20	Delete	Edit
daniel_jackson_20	Isolation null at time 20	Delete	Edit

Clique Count
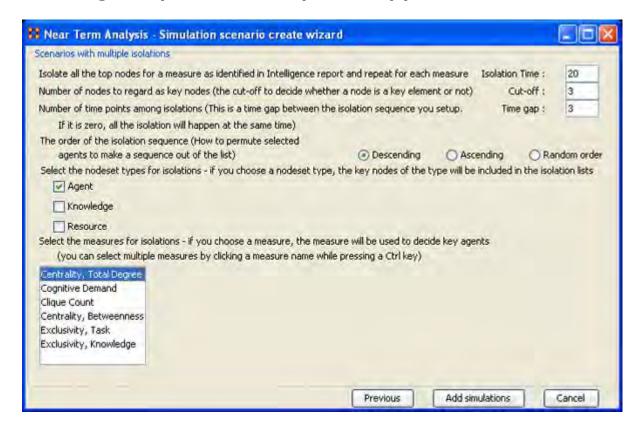
Isolation Case	Isolation Information	Delete	Edit
baseline	baseline		
col_jack_o'neill_20	Isolation null at time 20	Delete	Edit
ren'al_20	Isolation null at time 20	Delete	Edit
daniel_jackson_20	Isolation null at time 20	Delete	Edit

When you have run the measures proceed to **Analysis Results**.

Near Term Impact Option 4

A set of agents by a set of cases by measure(s).

Option 4 allows the addition of adding **Time gaps** between the isolations and whether to isolate them in descending, Ascending, or Random order.

Now proceed to **Analysis Results**.

Near Term Impact Results Menu

Save the Construct launch scripts : Information on Construct can be obtained at the following URL:
http://www.casos.cs.cmu.edu/project/construct/index.html.

Show text report on the ORA main window : Displays the report in Panel 3 of the ORA Main Interface.

Save the completely evolved Meta-Networks : Will save as an ORA Meta-Network any of the simulation cases with a checkmark in the **Select** column.

Save the html report : Navigate to a directory to save an **html** file that can be displayed in a browser.

Save the text report : Navigate to a directory to save an **text** file.

Save the Knowledge Diffusion line chart : Saves the chart as a **.png** file that can be used in other documents.

Save the Task Accuracy line chart : Saves the chart as a **.png** file that can be used in other documents.

Save the Energy Task line chart : Saves the chart as a **.png** file that can be used in other documents.
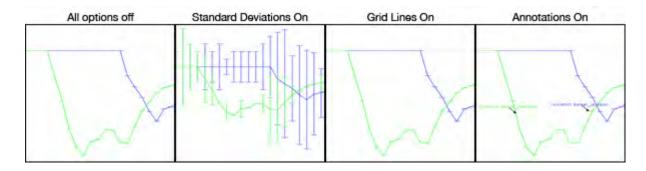
Save the Knowledge Diffusion result CSV file : Saves the results as a **.csv** file that can be used in other programs.

Save the Task Accuracy result CSV file : Saves the results as a **.csv** file that can be used in other programs.

Save the Energy Task result CSV file : Saves the results as a **.csv** file that can be used in other programs.

Close :

Copy the chart in the clipboard : Copies the window to the clipboard that can be pasted into other programs.
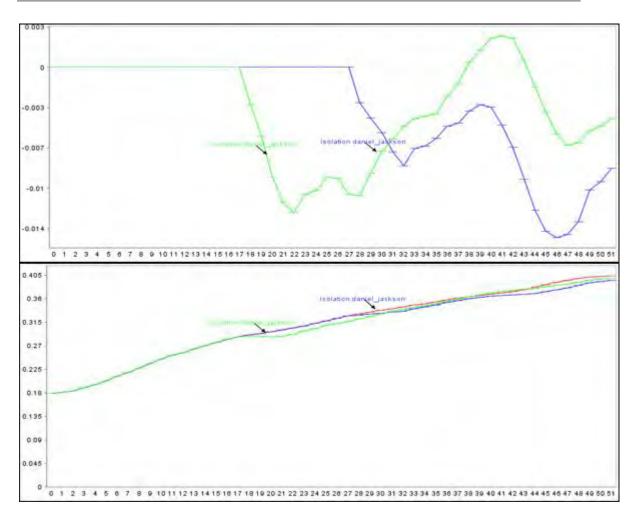
Standard Deviation ranges : **Standard Deviations*** displays the range of values for a particular instance.

Grid lines in the chart : Displays the grid lines for values.

Legends in the chart : Displays the legend underneath the chart.

Event annotations in the chart : Displays the event annotations within the chart.

Deviation from the baseline : Using the **baseline** numbers the chart shows the offset for the simulation.

192
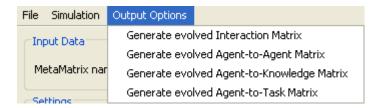
Show actual values : Uses he actual calculated values in the chart.

Background color : choose the background color of the chart of either **white or gray**.
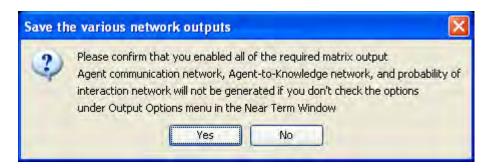
 Near Term Impact Results
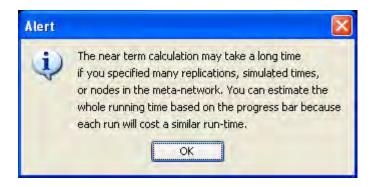
Now we're ready to Execute the simulations. But before that you must select the type of **Output Options** you want. From the Near Term Analysis menu select **Output Options ⇒ {any or all of the four options}.**
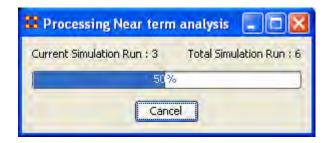
Now select the **[Execute]** button in the bottom-right corner. This will bring up two dialog boxes asking you to confirm you're selected output options and then an alert telling you that, this may take a while with a large Meta-Network and a lot of simulation runs.
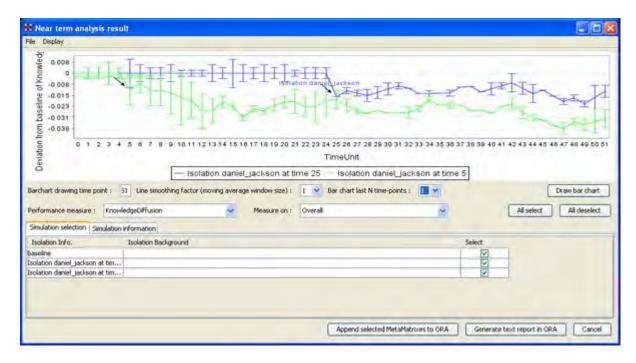
An alert box will display advising you that the near term calculation may take a long time depending on settings of your experiment.

After selected **[Execute]** the **Processing Near Term analysis** dialog will show you the progress.

Next will be displayed the **Near Term Analysis results** window.

The results of the experiment will compare against the baseline data established prior to the removal of any isolated agents. In the example above, we can see how the network was impacted in the time line when we isolated **daniel_jackson** and removed this node from the data set.

Further information can be obtained form the following references:

For knowledge diffusion and energy task

Schreiber, C. (2006) Human and Organizational Risk Modeling: Critical Personnel and Leadership in Network Organizations, PhD Thesis, Carnegie Mellon University, CMU-ISRI-06-120

For binary task accuracy

Lin, Z., K. Carley. 1997. Organizational response: The cost performance trade-off. Management Sci. 43(2) 217-234.

Visualization

Measure Charts – When run this function will plot the three graphs, **Bar Chart, Scatter Plot, and Histogram** on the chosen Network.

View Measures Over Time – Requires multiple Networks. First plots points for measure selected on each of the Networks active with the option to see a direct line from the earliest to the latest point.

View Network Over Time – Requires multiple Networks. Displays nodes and links for each of the active Networks.

Network Drill-Down– Allows for quick, plotting of nodes connected to a chosen node.

Node Cloud : Create an image with higher used concepts showing up in larger font sizes.

Color Grid : Creates a colored grid representing the networks. White cells represent no connections.

GeoSpatial Networks – Analyze, forecast, and visualize a wide variety of spatial interactions and networks within a global-scale. **See *GeoSpatial/Loom > GeoSpatial* for more information**

View Trails – Works with nodes moving between places over time. Requires the use of a trailset.**See *Loom* for more information**

View Networks

3D Visualization : Visualizes the selected network in the 3D Visualizer

2D Visualization : Visualizes the selected network in the 2D Visualizer

Path Finder : is used to find, not only, the shortest path between two nodes but also paths to other nodes when particular situations occur.

Sphere of Influence : Each node within a network has a unique Sphere of Influence or Ego Network. Essentially it's a direct relationship with it's neighbors as a function of specified path length. The ORA Visualizer allows you to focus on this relationship by creating an Ego Map centered on any particular node you choose.

Key Set Selector : uses information input by the user to create a simplified visualization.

Blank Visualization : Opens the Visualizer with no active Meta-Network. Allows you to create NodeClasses and Networks from scratch.
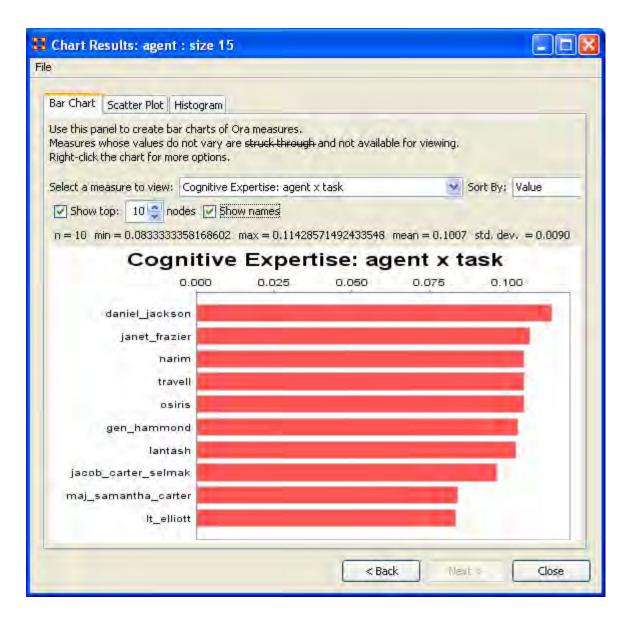
Charts Types

Below are examples of each type of ORA charts: **Bar Chart, Scatter Plot, and Histogram.**

Bar Chart

A bar chart contains horizontal bars representing the individual nodes in the set. These **rectangular bars** are usually proportional to the magnitudes or frequencies of what they represent.

Bar charts are used for **comparing two or more values**. The bars can be horizontally or vertically oriented. Sometimes a stretched graphic is used instead of a solid bar.
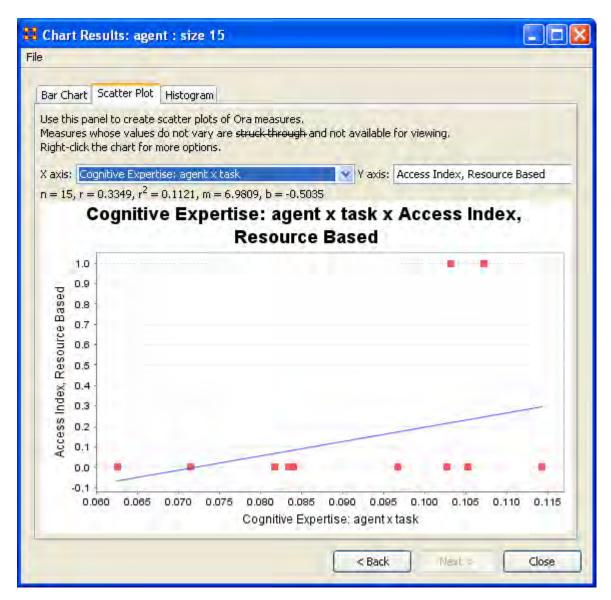
Scatter Plot

A scatterplot uses **Cartesian coordinates** to display values for two variables. The data is displayed as a **collection of points**, each having one coordinate on the horizontal axis and one on the vertical axis.

A scatterplot does not specify dependent or independent variables. Either type of variable can be plotted on either axis. **Scatterplots represent the association (not causation) between two variables**.

A scatterplot can show various kinds of relationships, including **positive (rising), negative (falling)**, and no relationship. If the pattern of dots slopes from lower left to upper right, it suggests a positive correlation between the variables being studied. If the pattern of dots slopes from upper left to lower right, it suggests a negative correlation. A line of best fit
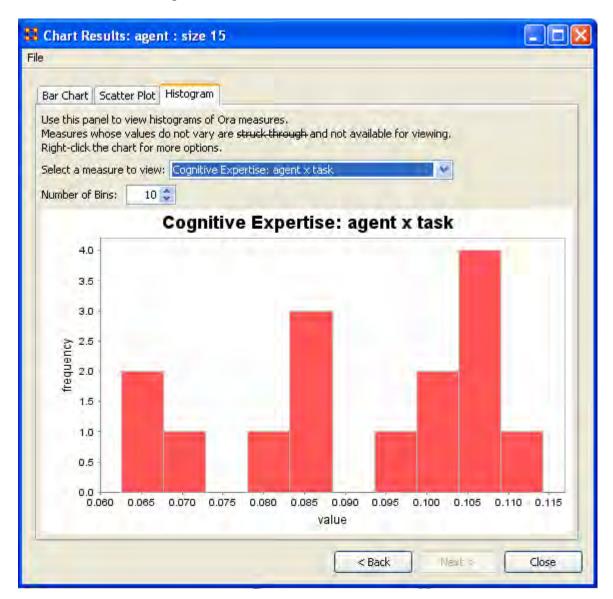
can be drawn in order to study the correlation between the variables. An equation for the line of best fit can be computed using the method of linear regression.

Histogram

In statistics, a **histogram** is a graphical display of tabulated frequencies. A histogram is the graphical version of a table which **shows what proportion of cases fall into each of several or many specified categories**. The histogram differs from a bar chart in that it is the area of the bar that denotes the value, not the height, a crucial distinction when the categories are not of uniform width *(Lancaster, 1974)*. The categories are usually specified as non-overlapping intervals of some variable. The categories **(bars)** must be adjacent.
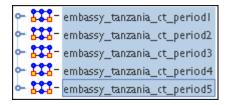
The word histogram is derived from histos and gramma in Greek, the first meaning web or mast and the second meaning drawing, record or writing. A histogram of something is thus, etymologically speaking, a drawing of the web of this something.

View Measures Over Time

Using the **View Measures Over Time** on a Network allows an analyst to see how a network measures change over multiple time periods. If you use non-dated data, ORA will alert you if your Meta-Networks are not dated.
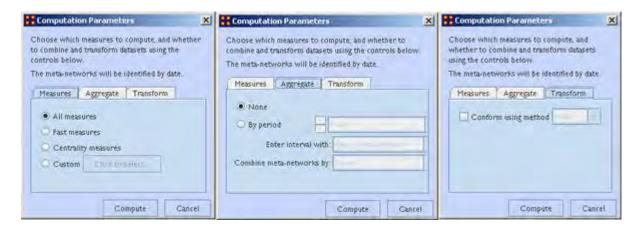
Highlight all the Meta-Networks and selectr from the main menu
Visualizations > View Measures Over Time...

NOTE : Measures Over Time will only be run on the selected Metw-Networks.
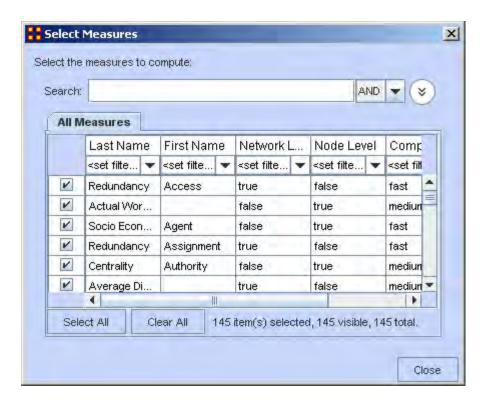
Computation Parameters

First, the Computation Parameters dialog window appears. Here you can select the measures to use and whether to combine or transform the datasets. When done, click the **[Compute]** button to continue to the analysis.

Measures tab : gives you control over the measures used in the computation. With the Custom option you can select as many or few measures to use.

- **All measures :**

- **Fast measures :**

- **Centrality measures :** Uses only the Network Centralization measures.

- **Custom :** Calls up a dialog box which allows you to selectively pick which measures to use. Place a check mark next to any measure you want to use. You can also use the [**Search**] textbox to find a measure(s).
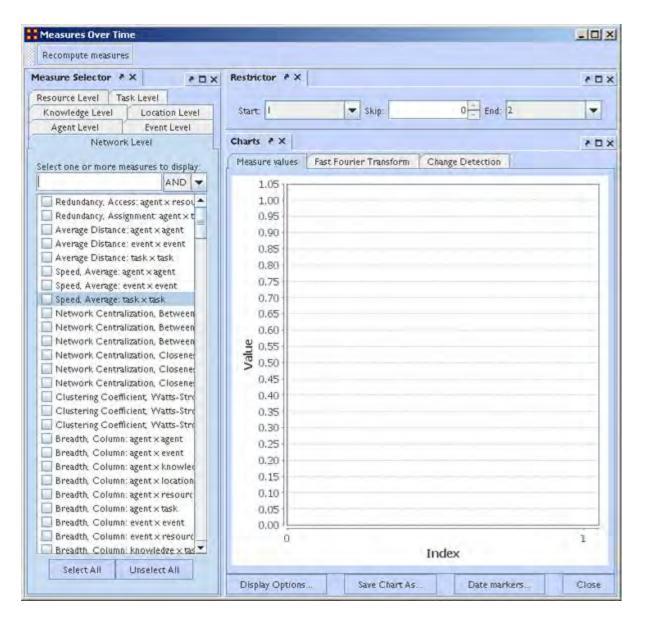
Aggregate tab : Combines multiple dates into one entry. If you have one meta-network for each day of the year, aggregating by month would reduce the 365 meta-networks into 12 meta-networks (the 12 would contain the 30 meta-networks for each day of the month unioned together).

Transform tab : The **Conform using method** makes the node classes indential across Meta-Networks. **Union** adds isolate nodes, and **Intersect** removes isolate nodes.

NOTE : Measure values computed on conformed Meta-Networks may differ from those computed on non-comformed data.

Measures Over Time window

This window is divided into three panes.
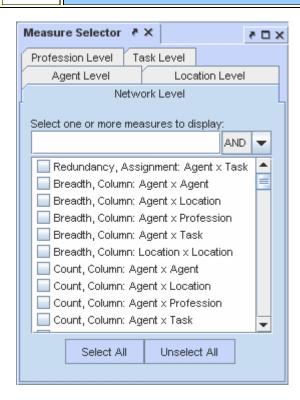
The Measure Selector Pane contains tabs for all NodeClasses in addition to a Network Level tab. If the **Network Level** tab is selected you can choose which measure(s) to apply to the entire Meta-Network. If a **NodeClass** tab is chosen you can choose which Node(s) to view and in the **Select a measure:** text box select the measure to apply to them. All actions are instantly updated in the **Charts Pane**.

The Restrictor Pane allows you to set the beginning and ending time periods and additionally you can skip any number of intermediary steps.

The Charts Pane contains three tabs and a legend at the bottom showing the colors associated to which nodes. Each pane is further described on their own pages.
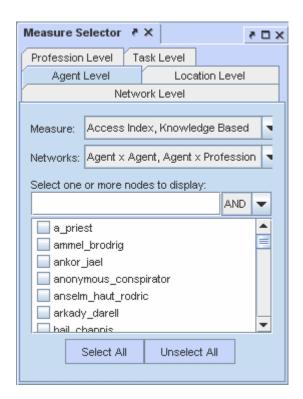
Measure Selector Pane

The Network Level tab is a list of all the measures used in the current operation. Place a check mark next to any measure you want to use.

The [**AND/OR**] drop down allows for boolen operations on the measures.

The [**Select All / Unselect All**] buttons can be used as a starting point to select mostmeasures or only a few measures.
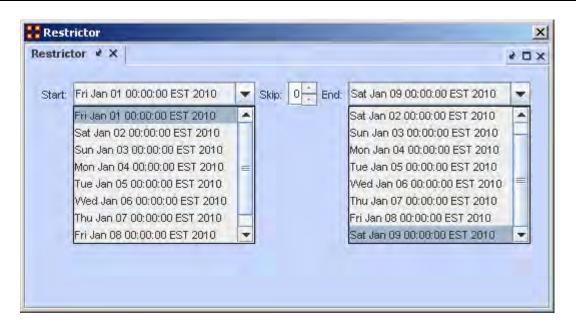
All the individual Node Class tabs work identically. Like the Network Level you place a check mark next to the nodes you want to use and it also contains the [**AND/OR**] drop down.

The **Measure:** and **Networks:** drop downs allow you to hone you selections.

The [**Select All / Unselect All**] buttons can be used as a starting point to select mostmeasures or only a few measures.
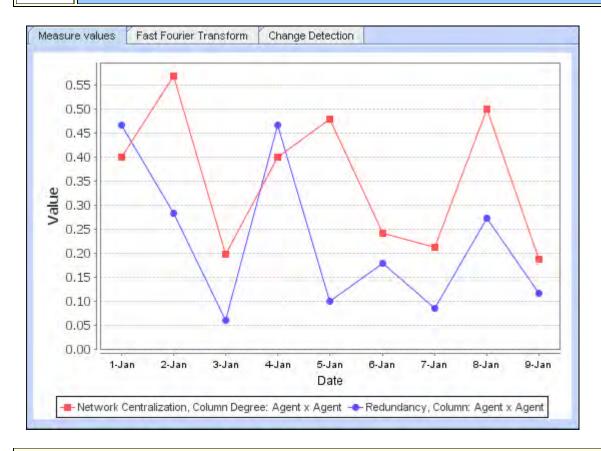
Restrictor Pane

The **Restrictor Pan** allows you to set the Starting and Ending Meta-Network to use and also to skip a number of Metaworks in the middle.
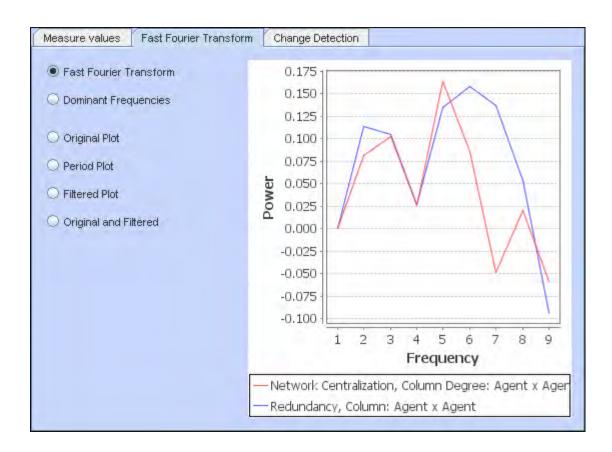
Over Time-Measures Tab

Over Time-Fast Fourier Tab

Frequencies

Fast Fourier Transform : Transform measures values to identify cycles in a temporal situation. Doesn't plot measure values. It plots strength of cycles. (i.e. traffic patterns could have a heavy/light traffic pattern or a weekly pattern, heavy on weekdays and light on weekends). Defines cycles of different lengths.

Dominant Frequencies : Selecting Dominant Frequencies on the radio button to the lower right, displays only the statistically significant frequencies.

Plots

Original Plot :
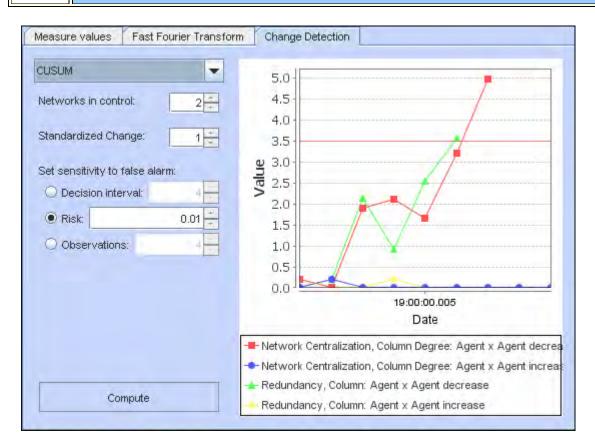
Period Plot :

Filtered Plot :

Original and Filtered :

Over Time-Change Detection Tab

CUSUM

Keeps track of cumulative total of deviation across time. No matter how small the change.

Networks in Control : Number of Networks to use as a control group

NOTE : If your change time is in the "Networks in control" bunch then you'll never see the change.

Standardized Change :

Set sensitivity to false alarm

Decision interval :

Risk : Higher risk numbers will possibly give more false alarms.

Observations :

EWMA

Exponentially Weighted Moving Average. Check Wikipedia. Incorporates historical data but places more emphasis on recent data.

Networks in Control : Number of Networks to use as a control group

Risk : Higher risk numbers will possibly give more false alarms.

Weight : Shows the weight of the newer observations over the older observations. Start and Finish is for which network to use.

Shewhart X-Bar

Show how different the current observation is from the baseline. This only takes into account the current observation.

Networks in Control : Number of Networks to use as a control group
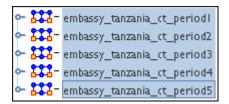
Risk : Higher risk numbers will possibly give more false alarms.
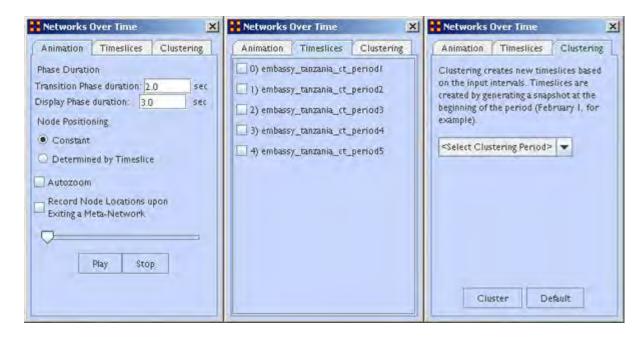
View Measures Over Time

View Network Over Time : Uses all selected Meta-Networks for analysis in various time frames. Select all the Meta-Networks to use in the analysis.

NOTE : It is of the utmost importance that the Networks in all Meta-Networks are named the same. If not, the Over Time function will not work correctly.

Now select from the menu **Visualizations > View Network Over Time** This will bring up the visualizer along with the **Networks Over Time** window.
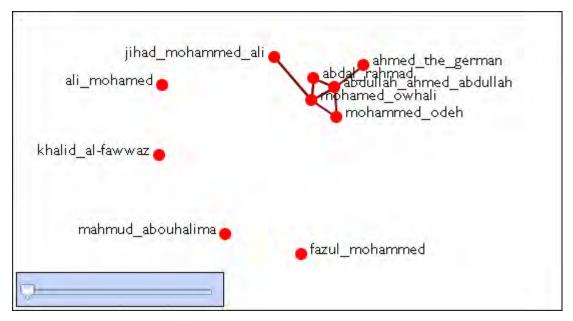
The Animation Tab controls the workings of the animation. It also contains the Play/Stop buttons.
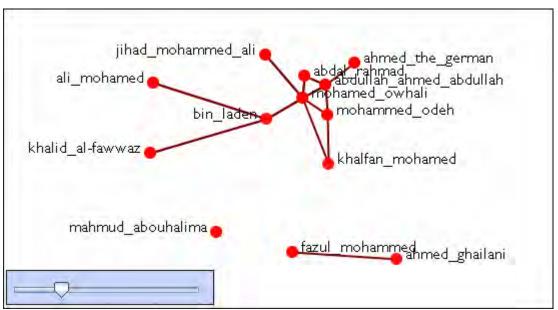
The Timeslices Tab contains check boxes for all active Meta-Networks. Placing a check mark in the boxes will use that particular Meta-Network.
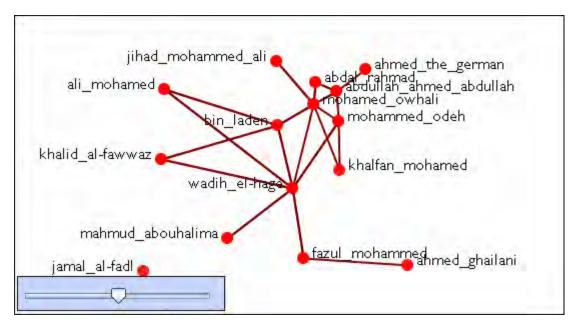
The Clustering Tab allows you to set the time frame for clustering by **day, week, month, or year**.
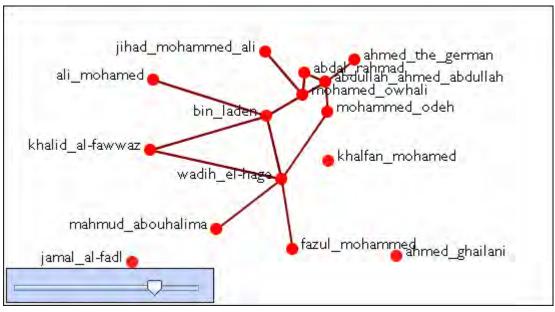
Animating the Display
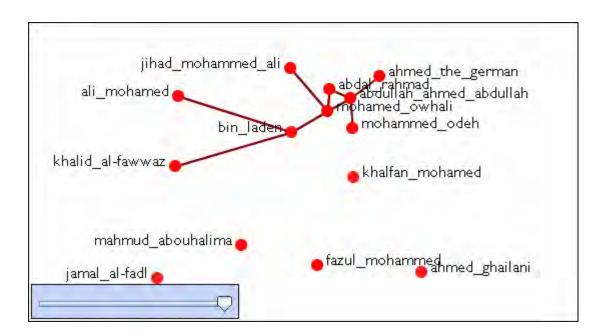
Pressing the **Play** button will start the animation. Below are the five **agent x agent** Networks. In different time slices new nodes appear and the connections between the agents changes.

211
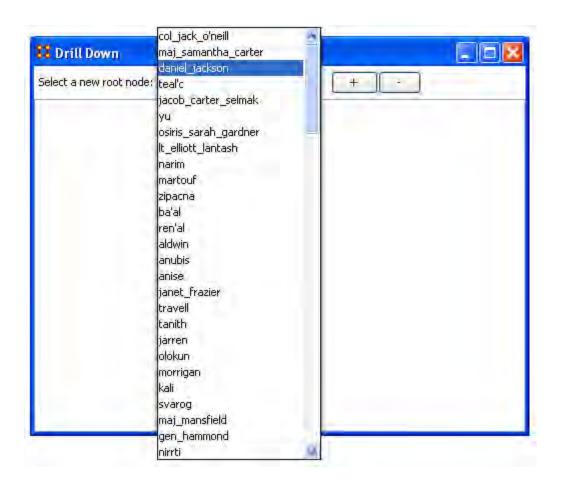
Drill Down

DrillDown allows for any path to be followed.
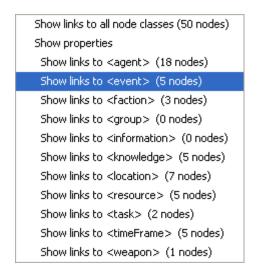
From the drop down textbox select the node you wish to start with. Here we'll begin with **daniel_jackson**.
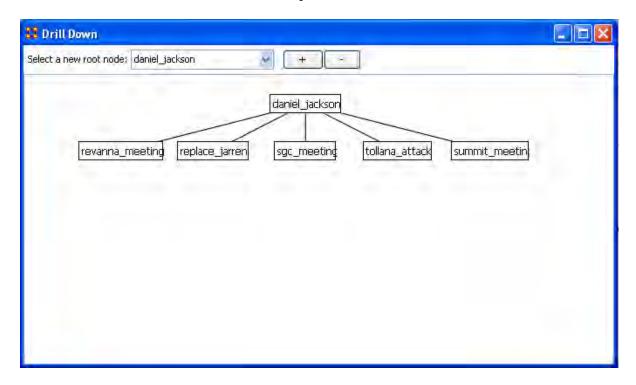
The dialog box below will appear confirming you want to discontinue your last drill-down session.
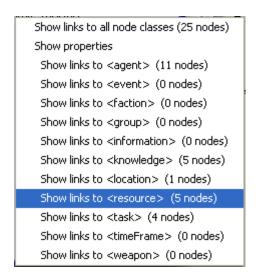
Click on the node and a contextual menu will appear with all the links it has to other nodes. I select **<event> (5 nodes)**.
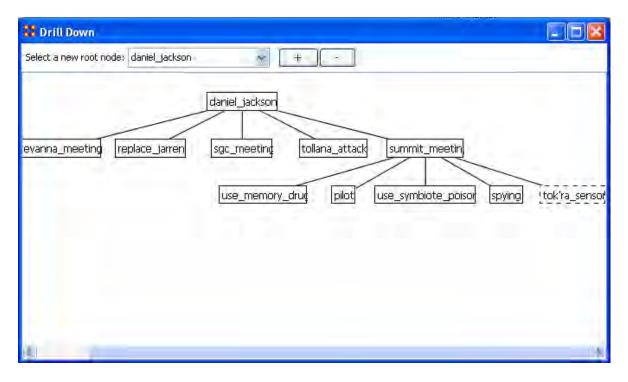
This shows the five events the daniel_jackson was at.

Next I select **summit_meeting** and from the contextual menu select **<resource> (4 nodes).**

It **drills down** showing the five links to resource nodes connected to the summit_meeting.
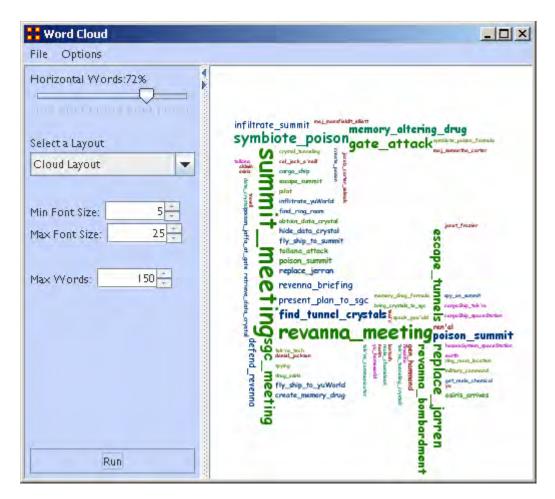
This can be continued for as long as you need.

The **[+]** and the **[-]** buttons increase and decrease the font size.

Node Cloud

 In ORA you can opt to visualize your data as a word cloud. A word cloud can loosely be described as a visual representation of words that appear within your data by displaying words that appear with the most common frequency as the largest. Below is a screen image of the SG1 data set as visualzed in ORA's word cloud feature.

Word Cloud Tools and Settings

File : By accessing the File Menu you can open a previously saved Meta-Network, Save an Image to the clip board, and close your Meta-Network word cloud.

Options : In options, you can choose the measure you wish to apply to your word cloud visualization.

Horizontal Words : You can use this slider to set the ratio of the horizontal words visualized to the vertical word display (e.g, changing this setting to 100% would result in all horizontal word visualizations and no vertical words. Conversely, changing this setting to zero has the opposite effect.

Select a layout : Choose either word cloud layout (default) or a square layout. The square layout will attempt to visual your word cloud in approximate equal ratio of width and height.

Min. Font Size : Sets the parameter for the smallest font size to display. Note: if you change this setting, click "run" to have the new setting applied.
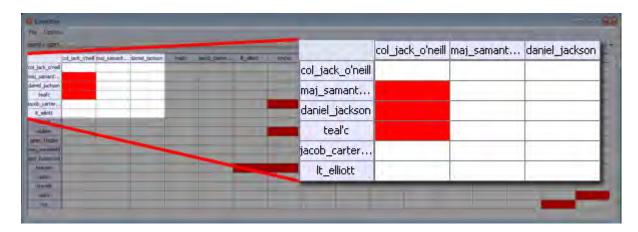
Max. Font Size : Sets the parameters for the largest font size to display. Note: if you change this setting, click "run" to have the new setting applied.

Max Words : Sets a limit on the maximum amount of words to visualize (e.g., if you set this at 10, the word cloud will only visualize the words that appear in the top ten in terms of frequency.

Run : Click **[Run]** to apply any changes to the settings.

Color Grid

The **Color Grid** is designed to allow you to quickly see what connections are in a network. Whereas the Editor either uses numbers in it's binary view or check boxes in it's binary view the striking red blocks make it easy to discern who has a connection to whom.

Menus

File > Save Image As... Creates a color image (**JPG or PNG**) of the currently display grid.

	mathemat...	attorney	judge	politician	spy
hari_seldon	■				
gaal_dorn...	■				
jerril					■
lors_avakim		■			
linge_chen			■	■	

File > Save Image As... (Greyscale) Creates a greyscale image (**JPG or PNG**) of the currently display grid.

	mathemat...	attorney	judge	politician	spy
hari_seldon	■				
gaal_dorn...	■				
jerril					■
lors_avakim		■			
linge_chen			■	■	

Options > Toggle Headers : Toggles on/off the display for the headers and reduces the size of the grid.
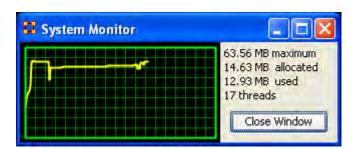
Help Menu

The Help Menu contains information which is beneficial in running ORA.

Help Contents : Calls up this Help File System

System Monitor :

218

About Ora : Contains information regarding this software, the institute of origin, permissions and warranties, version and build dates, and Java usage.
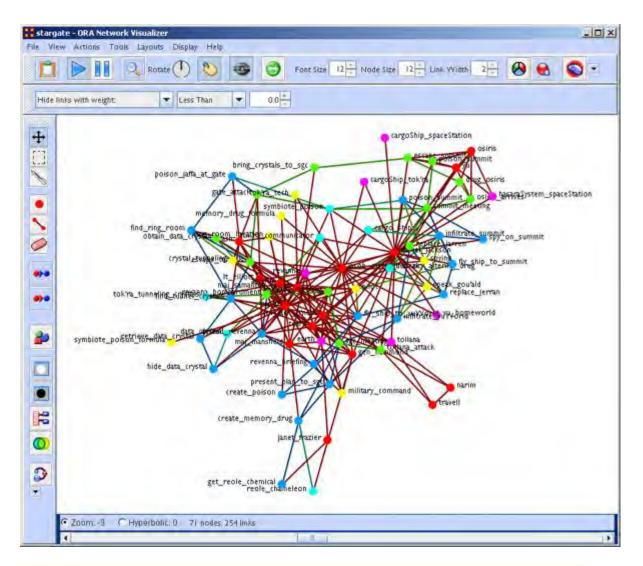
The ORA Visualizer

The **ORA Visualizer** produces a graphical model of a Meta-Network comprised of Links and Nodes. Moreover, you can interact with your Meta-Network model by **removing key actors, isolating certain links, focusing on any particular relationship between two nodes using tools such as the Path Finder.**

You can apply a host of grouping algorithms, look at node's Sphere of Influence, Locate Nodes, which may be hidden in complex clusters of nodes and links.

The Visualizer Interface

Below is a visualized Meta-Network displayed with the current tools available in the Visualizer. In this view all nodes and links are visualized. Note the different palette areas and options available from the Visualizer.

Closing the Visualizer

When you close the Visualizer you will be asked [**Save Visualizer options?**]. Selecting [**Yes**] will maintain the current state this Meta-Network is in. i.e. If you have hidden any nodes they will not appear the next time this Meta-Network is opened.

Synchronization

Synchronization allows you to work in the Editor and instantly see those effects in the Visualizer (or visa versa). Adding or removing a link in the Editor will place that link on the screen in the Visualizer.

The ORA Layout Algorithm uses a standard spring embedder algorithm. For large data sets, this has been enhanced using the Barnes-Hut Simulation for calculating the force of repulsion between any two nodes. **Reference :**Nature 324, 446 - 449 (04 December 1986); doi:10.1038/324446a0, A hierarchical O(N log N) force-calculation algorithm, Josh Barnes & Piet Hut

Visualizer Basics

Contained on this page are some basics to the visualizer. Many are small, yet important aspects to the visualizer.

Link Thresholds

ORA attempts to find the link underneath the cursor. But as the number of links increases it becomes more difficult for to acquire the correct link.

A threshold limit of **5,000** has been set. ORA will not attempt to locate the link under the cursor when viewing a Meta-Network with more than this number.

The Visualizer Toolbar

This section explains the function of each of the toolbar's **default** buttons. The user can add or remove buttons from the toolbars. These are explained in other sections of this help.

When you first load a Meta-Network you will see the **pause button** depressed in the Visualizer tool bar. You can **layout** the Meta-Network by clicking the **play** button. The visualization will space apart across the screen. This is called **laying out** a Network. The Visualizer is separating nodes and links that, at first, appear on top of each other. You can **pause**

this process again when you think the Meta-Network is satisfactorily spaced apart or you can let ORA lay out the entire Meta-Network. Your computer speed, as well as the complexity of Meta-Network, will determine how long this takes.

Copy / Paste : This feature allows you quickly capture a visualization and save it to be pasted into another document later.

Play / Pause : The **Play Pause** function works very similar to the play pause feature on any device. The pause button **(two vertical bars)** is depressed, the Visualizer stops laying out a Meta-Network. When the play button is depressed **(right pointed triangle)** ORA Visualizer begins laying out the Meta-Network as described above. The red ellipse highlights on the tool bar where this feature is located and accessible.

Magnifying / Maximizing : The **Magnifying Glass** icon with the plus sign inside it, allows you to instantly fill the Visualizer window pane with the currently rendered Meta-Network. The red circle in the screen shot above highlights where to access this feature on the Visualizer tool bar.

If there are any nodes off screen, clicking the button once will bring all the nodes into view on the screen.

NOTE : This function works well when selecting a small part of your overall Meta-Network and magnifying it to fill the visualizer screen.

Rotating The Visualization : To **Rotate** your visualization look for the word next to the **Noon-pointing sundial** icon in the tool bar. This feature is another way to manipulate your visualization to fill the entire visualizer window pane.

NOTE : Use this feature in conjunction with the magnifying glass function to work your visualization into largest size possible.

Show Labels : The **Show Labels** button allows you to toggle the labels on & off in the visualizer. Default is is **ON**.
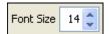
Show links : The **Show Links** button allows you to toggle the links on & off in the visualizer. Default is is **ON**.
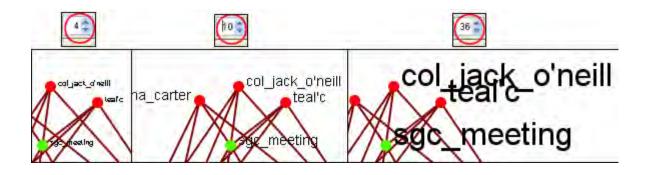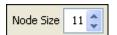
Show Arrows : The **Show Arrows** button allows you to toggle the arrows on & off in the visualizer and show directional data. Default is is
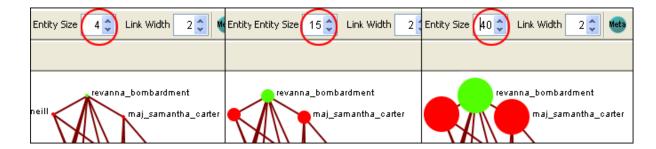
OFF.

Font Size : The **Font Size** allows for setting the size of the font in the visualizer. This allows you to change the size of the font to suit you needs for each individual visualizer set. It ranges from **4**, which is almost unreadable, to extremely large numbers, which can also be unreadable by virtual of overcrowding. So pick the size of font that best suits your screen.
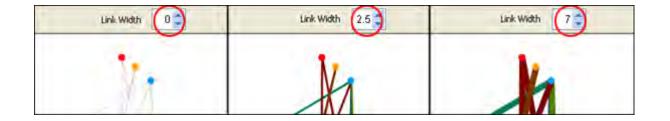

Node Size : The **Node Size** allows for changing the node size, larger or smaller by increments of one. Four is the smallest size possible and creates a very small, almost indistinguishable dot. As the number increases is becomes easier to see the individual nodes until the number gets to a point where is can simply overpower the screen. But this gives you the ability to customize the entity size to suit your needs.
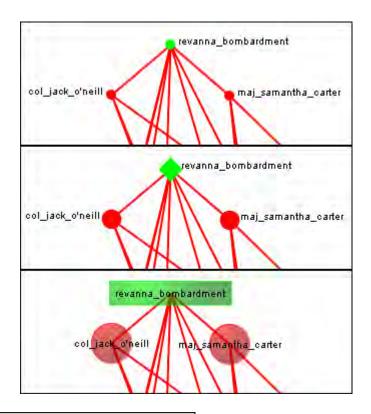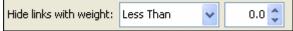
NOTE : This also affects the size of the nodes in the Legend.

Link Width : Allows for changing the general width of the line, larger or smaller. And as the options before you can pick the exact size which best suits your needs for the display.
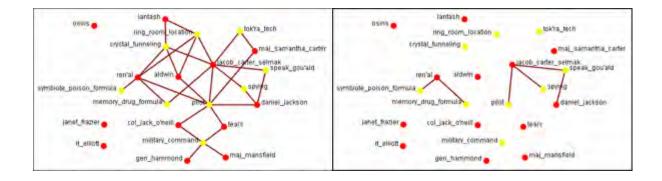
 **Meta-Node Size :** Use the **Meta Node Manager** to handle creating and removing Meta-Nodes.

 Node Sizes : The **Toggle Node Size** button rotates through three sizes for the nodes **(as seen below)**.
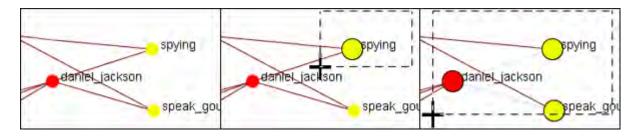

Hide Links by Weight : Use the **Hide Links with Weight** to select which links to display. As long as there are links with different values you can hide links of smaller values leaving only the more important links visible. This can assist in seeing links that might normally be too cluttered to view. Below left is a Meta-Network with all links in view. Below right is the same Meta-Network with links less than 2.0 hidden.
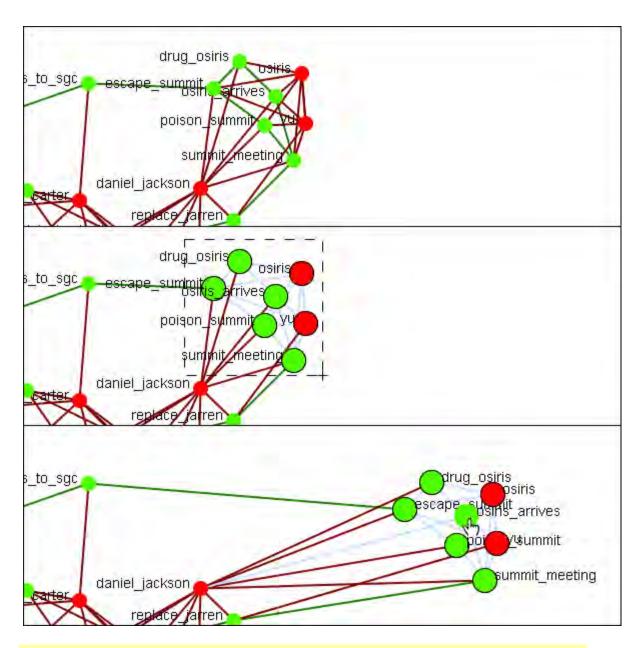
![move cursor icon]

Drag : Used to move the view around the Meta-Network to see different parts. Press and hold the left mouse button and move the pointer around to view various parts of the Meta-Network.

Marquee : Click and drag the marquee over nodes to select them.

After releasing the mouse button the set of selected nodes will remain enlarged enclosed in a dashed line.

After selecting a group of nodes they can be moved all together. Press the **[Pause]** button on the toolbar, switch to the **[Move]** tool, then grab and move the group. This is useful in creating displays of small portions of the visualizer.

The Paint Tool : The **Visualization Paint tool** allows you to draw on the visualization. You can select a brush size and a brush color via a small pop-up menu that appears whenever paint mode is selected. Nodes are also selected whenever you stop drawing, using the shape drawn (plus a line from the endpoints of the drawing) as a giant polygon, which selects nodes based on whether or not they're inside the polygon. There's also a "clear" button that removes all paint. "painting" works similar to using a marker on your computer screen – no matter how much you zoom/pan/do whatever to the visualization, the marks

will remain in the same place.

Node Creator : You can create new nodes within pre-existing NodeClasses or create them in a new NodeClass.

Link Creator : Click on one node then drag the pointer to a second node. A link between the two will be created.

Node / Link Eraser : While the Eraser tool is active any node or link clicked will be removed from the Meta-Network,/tr>

Visualizer Toolbar

To help facilitate your workflow ORA allows you to mount/unmount tool buttons in the Visualizer's toolbar. The Visualizer contains two toolbars for this function. One on the top and another along the left side. Either of these toolbars can be placed in the horizontal or vertical position. Click the down-triangle to open the dialog box.

The grey-out buttons are the default set. The buttons at 100% are the ones I added to my GUI.

227

The **Configure Toolbar** dialog box will appear. You will notice that some items are already checked. These correspond to the buttons already present. Placing a check mark next to any tool instantly mounts it on the toolbar. Removing the check mark from a button instantly removes the button from the toolbar.

When you are finished click [**Close**].

If at any point you can select the [**Restore Default**] button which will remove all the changes you made and restore the toolbar to the buttons found in the image at the top.
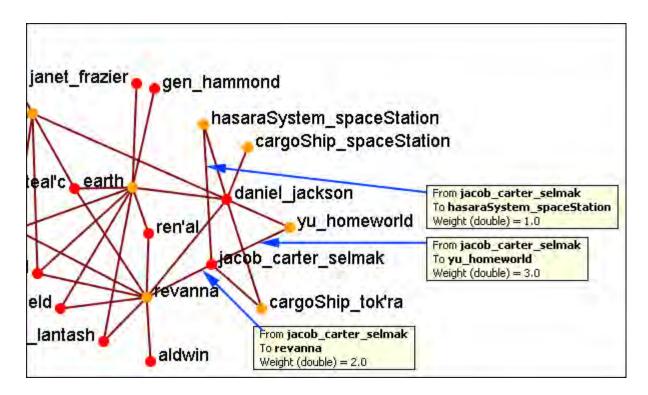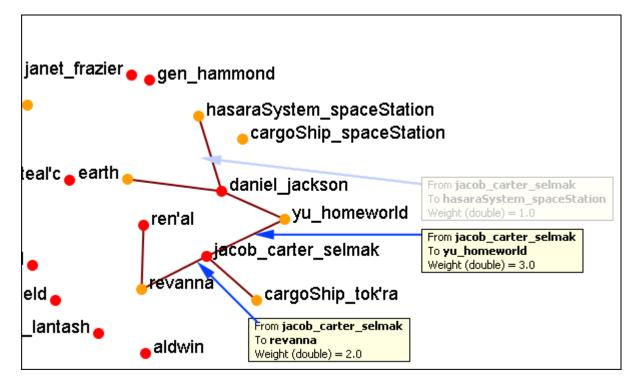
Hide Links By Weight

ORA can visualize your network with only the more **weighted** links displayed. Below are procedures to accomplish this task.
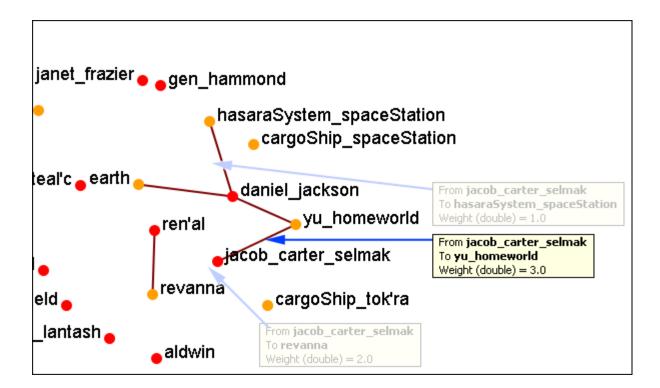
You can select to hide links either **Less Than or Greater Than** a certain value. Below is an agent x location network. And there are three tooltips displayed for jacob_carter for the values of 1.0, 2.0, and 3.0. We'll keep the hiding style as Less Than.

First we'll increase the value to 1.1 hiding all lower values. The link to the **hasaraSystem_spaceStation** is removed from view. The others, however are still on the screen.

Now let's increase the value to 2.1. This removes the link from jacob_carter to revanna.

If we increases to value to 3.1 then all of the links would be removed.

In this way you can remove links which are of lesser important to concentrate on a smaller set of data.
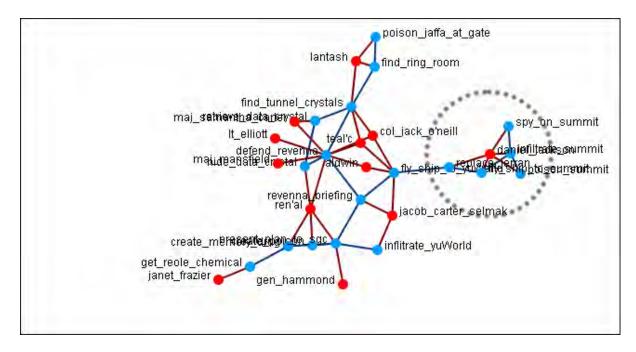
Hyperbolic View

The Hyperbolic function creates a **bulge** within your visualization adding a sense of depth. By moving the slider from right to left, you can increase or decrease this bulge effect and create different depth-added views. By moving the network around the window you can create the view you need. The **Hyperbolic radio button** and **slider** can be found at the bottom of the window.

This tool is useful when nodes may be very tight on screen such as the ones in the following images. Using the Hyperbolic tool you can help space them out for easier viewing. Here is how to do so:
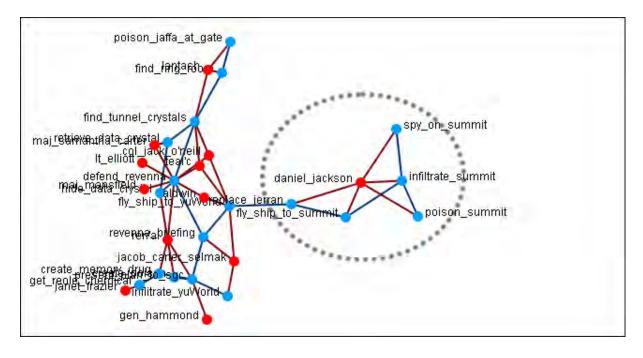
Select the Hyperbolic tool and move the slider to the right. Then click-and-hold on some part of the white background and move the entire network to the left. You will notice the space between the nodes begins to increase the closer them come to the center of the screen.

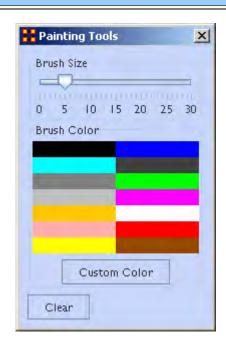Now the nodes are able to be read much easier.

Visualizer Paint Tool

The **Visualization Paint tool** allows you to draw colored lines on a visualization. You can select a brush size and a brush color via a small pop-up menu that appears whenever paint mode is selected.

Nodes are also selected whenever you stop drawing, using the shape drawn (plus a line from the endpoints of the drawing) as a giant polygon, which selects nodes based on whether or not they're inside the polygon. There's also a **clear** button that removes all paint. Painting works similar to using a marker on your computer screen - no matter how much you zoom/pan/do whatever to the visualization, the marks will remain in the same place.

From the Paint Palette you can select a **brush size** by adjusting the slider. Then choose a color by clicking a pre-set colour or clicking the **[Custom Color]** button and select any color. Clicking the **[Clear]** button sets everything back to default. Once brush size and color are selected, you can draw on the Visualization.
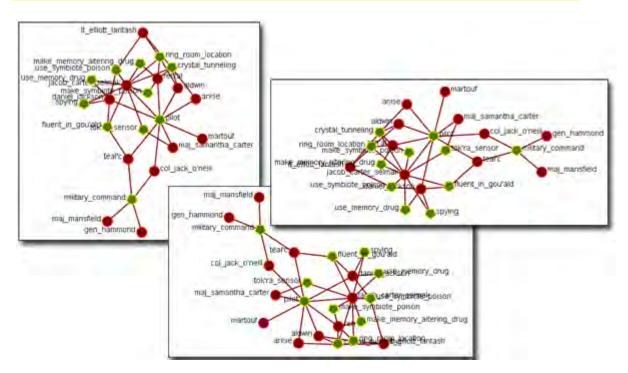
To undo your drawing click the **[Clear]** button.

Rotating A Visualization

You can rotate a visualization with the slider bar at the bottom of the interface or from the tool bar at the top. To do this from the slide bar, click the rotate option toward the bottom of the interface. This activates the rotate feature.

As you move the slider from the left to right, your visualization will rotate correspondingly. From the tool bar, you can click inside the rotate icon, which will turn the visualization depending on exactly where you clicked.

In the screen shot below, the yellow ellipses highlight where to access ORA's rotate tool both at the bottom and top of the visualizer interface.

NOTE : Use the rotate feature in tandem with the magnifying glass to find a visualization's maximum screen size.
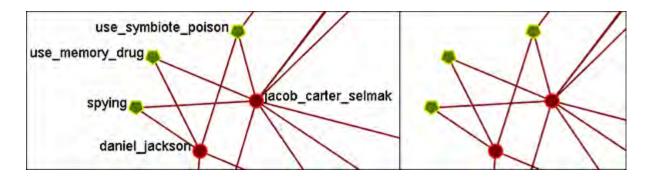
Show/Hide Labels

Node labels, which often prove helpful in describing Networks, can sometimes clutter a visualization. Thus, it may be necessary to eliminate labels from your visualization. To remove labels go to Visualizer tool bar and find the **Show Labels** button.

This initial screen shot shows the visualizer with all the labels.

When you click the Label button on the Visualizer tool bar ORA toggles the labels on and off. Note that now a much more clearer conceptual picture of the network is produced. You can flip between the two views.
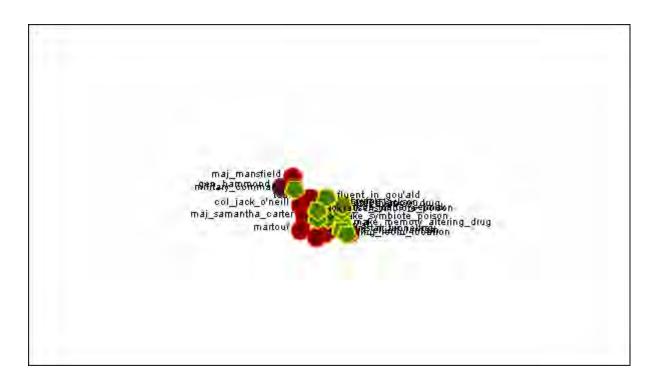
Zooming

At the bottom of the Visualizer interface is a sliding zoom bar (image below).
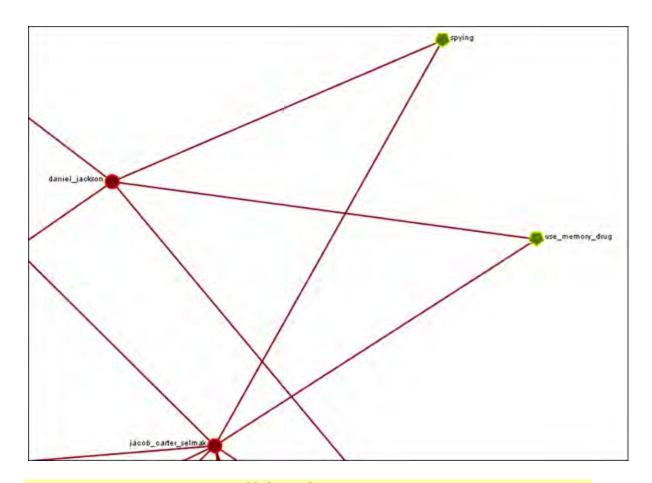
You can drag the scrollbar left **(to zoom out)** or right **(to zoom in).**

In the screen shot below the zoom bar has been set to the far left, the absolute zoom out possible. At this point very little is distinguishable.

In the next shot the zoom bar has been placed almost to the far right, absolute zoom in. It allows for extreme closeups of various parts of the Meta-Network.

File Menu

Open Meta-Network : This opens an **Open File** dialog box so you can navigate to a directory and open a Meta-Network into the Visualizer.

Open Ora-Image :

Save Image to File : This opens a **Save As...** dialog box and allows you to save an image of the current Visualizer in one of the following formats: **PNG, JPEG, SVG (scalable vector), or PDF**.

Save Meta-Network to File : After you've made changes to the information you can save the current status of the Meta-Network into a new file in a **DyNetML*** format.

Save Special :

Save Meta-Network (Only Selected Nodes) : Saves a new Meta-Network file which will contain only nodes currently selected, i.e. a subset of the currently loaded Meta-Network.

Save Grouping to New Meta-Network :

Save Meta-Network (No Dotted Links) :

Save Meta-Network (Only Dotted Links) :

Copy Image to Clipboard : Similar to the **Save Image to File...** only saves it to the clipboard where you can paste it into an image-editing program.

Add Meta-Network to ORA : Save the currently **visible** network into a new Meta-Network titled, **New Meta-Network**. This new Meta-Network can be renamed accordingly.

Add Grouping to Current Meta-Network :

Add MetaNodes to Current Meta-Network :

Visualize in 3-D :

GeoSpatial Networks :

Node Cloud :

Close : Close the Visualizer and returns to the Main Interface.

Save Image to File

An image can be captured from the Visualizer and saved in one of four formats:

- **PDF :** This format us usable with Adobe's Acrobat and Reader programs.

- **JPEG :** A standard compressed format for image editing.

- **PNG (**Portable Network Graphics). PNG is a bitmapped image format that employs lossless data compression. PNG was created to improve upon and replace the GIF format.

ORA Images are functional in regards to PNGs. Whenever you save a PNG, relevant information about the visualization gets stored inside the image. This makes saving a little longer, but not so long that you'd notice if you were not looking for it. These images **Node Colors**, **Node Location**, **Screen Center**, and **Zoom Value**.

Loading images into a visualizer will re-create the image inside the visualizer. Loading an image is as simple as **dragging-dropping** it onto the visualizer.
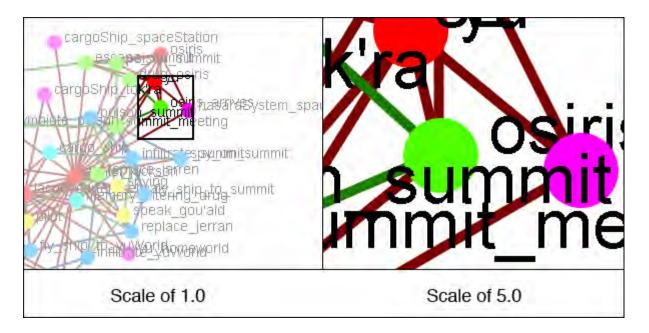
SVG : Stands for Scalable Vector Graphics. SVG is an XML specification and file format for describing two-dimensional vector graphics, both static and animated. SVG can be purely declarative or may include scripting. Images can contain.

Selecting Image Resolution on Save

After choosing a format form the drop-down menu and specifying a filename press **[Save]**. A dialog box will appear asking for the quality of the image. The **1.0** saves the image in the same resolution. A higher number will result in a higher resolution image.

Below are examples of an image saved with a scale of **1.0** (on the left) and then saved with a scale of **5.0** (on the right). Use a number higher than 1.0 if there is a need for a higher resolution image.

Scale of 1.0   Scale of 5.0

Views

This section of the help contains a description of the functions of the **View Menu**:

Change Font : Allows you to change the GUI font to any font on your system. Be careful with the font chosen as some fonts are built more for custom design rather than readability.

Log Hidden Nodes to Event Log :

Log Hidden Nodes to File :

Actions

This section of the help contains a description of the functions of the **Action Menu**:

Undo : Undoes a variety of actions performed in the Visualizer.

Add Node(s) : Function for adding Nodes directly into the Visualizer. These can be added to existing NodeClasses or within newly created NodeClasses. You also have the option of the number of nodes to create.

Add Node (with links or attributes) : Allows you to add a Node directly into the Visualizer. In addition you can create links and assign weights to these new links. You can also select attributes and assign values to these attributes.

Add Link : Allows you to create new links in a Meta-Network in the Visualizer. Then select the source and target nodes and then select the network. You have the option of creating a new network for this link. Lastly you can select the weight of the new link.

Notes on Display of Multiple Edges : If your Meta-Network contains graphs with **multiple edges between two nodes ORA will display the edges as curved lines so as to allow visualization of overlapping Edges. This is to say that if there is represented two nodes with four edges coming from four different graphs.**

Please note that **Tool tips are currently handled by math looking at the edge as if it were a straight line**. Since our line is not straight anymore, it will only be highlighted if the cursor's in the middle - nowhere near the edge. But otherwise, you should still see tool tips.

Delete Isolates... : Allows you to remove Isolate nodes in two ways.

Delete Pendants : Allows you to remove Pendant nodes in two ways.

Delete Currently Visible Isolates/Pendants : Currently Visible is exactly as it sounds. If it has no visible link on the screen, it's an isolate. It does not matter if there's a link that is hidden, it's an isolate/pendant.

Delete Underlying Isolates/Pendants : Underlying ignores was is currently set to visible and hidden. It looks at the data itself. If the Data says it's an isolate/pendant, then it's an isolate/pendant and is removed.

 Pin All Nodes : Set all nodes as unmovable

 Unpin All Nodes : Release all nodes and allow them to be moved independently.

Show Motifs

 Show Isolates : Nodes which have no connection to any other node.

 Show Dyads: Nodes which have a connetion to only one other node

 Show Triads : Nodes which have a connection to two other nodes

 Show All-In-All Cliques : A group in which every node is connected to every other node in the group

 Show Stars : A central node with connections to all other nodes in the group. The pendant nodes are not connected to each other.

Show Long Chains : Seeks out links which are nodes with only two neighbors.

 Show Cycles : A cycle exists if for any node in the group, there is a path back to that same node.

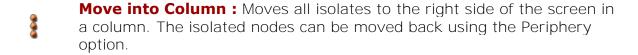**NOTE :** Cycles is the only motif that really looks at link direction.

Isolates

Hide Isolated Nodes : Remove any node which has **zero** links connecting it to other nodes.

Periphery : Moves Isolates back to their original position.

Combine into MetaNode : Collects all the Isolate nodes into a Meta-Node.

Move into Column : Moves all isolates to the right side of the screen in a column. The isolated nodes can be moved back using the Periphery option.

Hide Pendant Nodes : Remove any node which has **only one edge** connecting it to another node.

Recursively Hide Pendants : This function will run the both the **Hide Pendant** routine until there are no more Pendants.

Recursively Hide Pendants and Isolates : This function will run both the **Hide Pendant and Hide Isolate** routines until there are no more Pendants or Isolates.

Hide Nodes by Degree: You are asked for a **degree** (visible neighbors) a node has and will hide all of these.

Recursively Hide Nodes by Degree: Works similar to **Hide Nodes by Degree** but continues processing until no mode nodes can be hidden.

Show All Nodes: Any nodes that were previously hidden will be revealed.

Show All Nodes Except Isolates : Reveal any nodes that were previously hidden with the exception of nodes that have no links connecting them to another node.
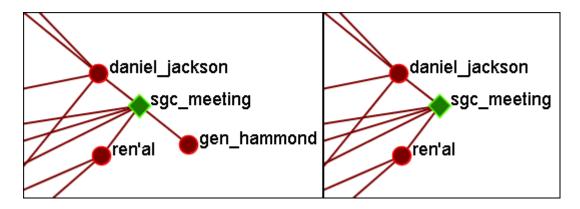
Compute Standard Measures for Visible Network : Compute Standard measures computes the following collection of measures on the visible network: Total Degree Centrality, Betweenness Centrality, Eigenvector Centrality, Cognitive Demand, Resource Exclusivity, Task Exclusivity, Knowledge Exclusivity. The measure values can then be viewed within the Entity Status Window.

Hide Pendants

Pendants share links with only **one** other node. Their linkage is tangential and therefore isolated from the core linkages you may be interested in examining. **Pendant Nodes**, like **Isolated Nodes**, can be hidden.

From the drop down menu in the Visualizer select **Tools > Hide Pendant Nodes**.

Below is a **Before/After** screen shot of the visualization before showing the removal of the pendant nodes.
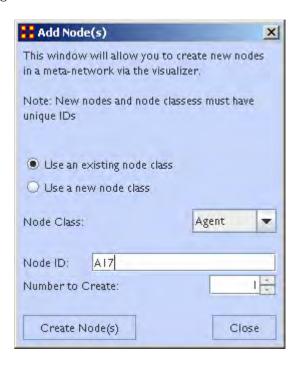
Add Node

IMPORTANT NOTE : Changes done using Add Node(s) and Add Link(s) are transient. Be sure to use either the **File > Save Meta-Network As...** (to save the entire Meta-Network with additions) or **File > Add Meta-Network to Ora** (to create a new Meta-Network using only the currently visible nodes). Changes can not be saved back to the currently open Meta-Network.

The **Add Node(s)** function allows addition of nodes to the current Visualizer. These **added Nodes** are not added to the currently open Meta-Network. If there is a need to save the Meta-Network with the added nodes then use the menu item **File > Save Meta-Network to File...** and create a new Meta-Network.

Below is the NodeSet for **Stargate** with the original 16 node NodeSet.

From the main menu select, **Actions > Add Node(s)** which brings up the
Add Node(s) dialog box.
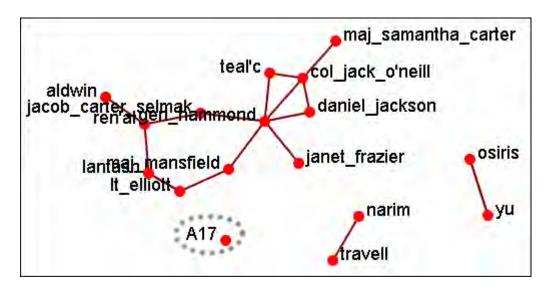
1. Use an Existing nodeset :

   allows for for a NodeSet selectable in option 2 or **Use a new
   nodeset** creates a new empty NodeSet.

244

2. This dropdown gives the option for selecting a pre-existing NodeSet to place the new node.

3. The **Node ID:** textbox is used for giving the new node an ID.

4. The **Number to Create:** specifies the number of new nodes.

After inputting the information to add node A17, the Visualizer will now display the new node as **A17** with no connections. In order to display this node with a **title** will require saving the Meta-Network and changing the information within the ORA Editor.
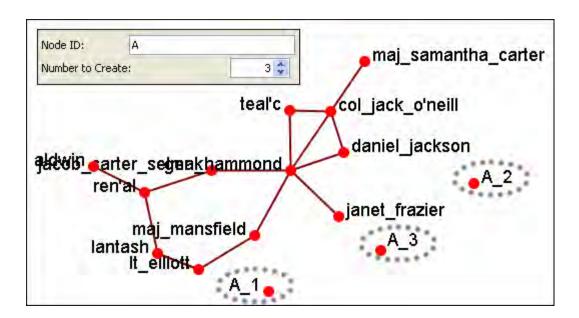
Adding Multiple Nodes
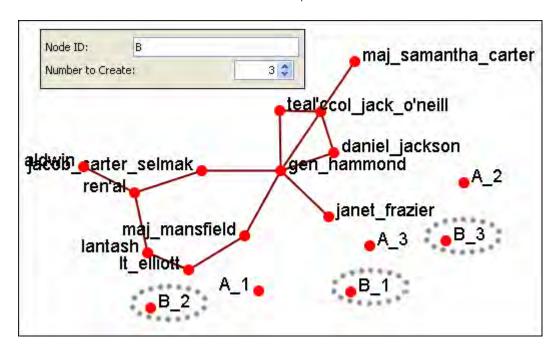
The procedure for adding multiple nodes is similar to adding a single node. The exception is the text in the **Node ID:** textbox. When adding multiple nodes only use a prefix. ORA will attach numbers, starting at 1, for each node.

Here's the addition of three new nodes and the prefix **A** will be used. Three nodes, **A_1, A_2, and A_3** are created on screen.

To add additional nodes use a **different** prefix.

Add Link

Allows you to create links between nodes in the Visualizer and place them in an existing NodeClass or create a new NodeClass. You can also set the weight for the new link.

1. **Start node :** The start of a directed edge.

2. **End node :** The end of a directed edge.

3. **Insert Nodes as they're clicked :** Fills in the text boxes for **Node**. The first click adds the node in the **Start node** section. The second click adds the node to the **End node** section. Any further clicks will make the previous End node the start node and the newly clicked node becomes the new End node.
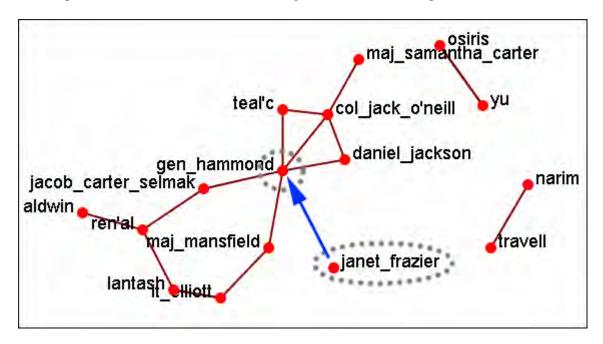
4. **Use an Existing network :** will place the links within the chosen network form the dropdown menu and **Use a new network** creates a new network from the name in option **5**.

5. **New Network ID :** is the name for the new network from option **4**.

6. **Network :** allows you to choice the Network to create the link in.

7. **Link Weight :** The weight to assign the edge between the nodes.

NOTE : Since this function creates **directed links** a second link would need created for reciprocal action.
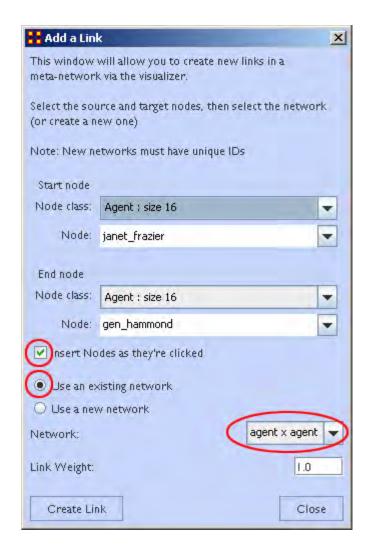
    Example : If a connexion was made starting with A to B, a second connexion would need to be made starting with B to A.

Adding a Missed Link
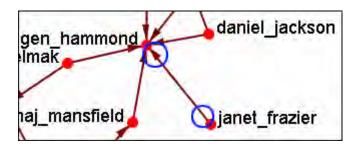
If after the Meta-Network was constructed and it's noticed an link is missing (e.g. janet_frazier should have been connected to gen_hammond), the missing link can be added immediately without returning to the editor.

From the Visualizer menu select **Actions > Add Edge(s)**.

Be sure that the **Insert Nodes as they're clicked** has been check marked and that **Use an existing network** is selected with the correct network is visible in the dropdown. Then select the **[Create Edge]** button.
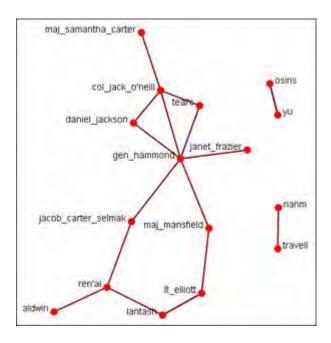
The two nodes are now connected with a **directed link**.

NOTE : this created a directional link from **janet_frazier** to **gen_hammond** but not the other way. To make it go both ways, a second link from **gen_hammond** to **janet_frazier** would need to be created.
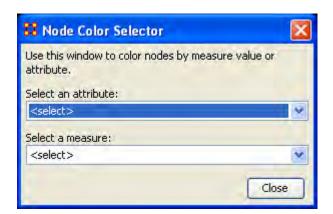
 Color Nodes By Attribute
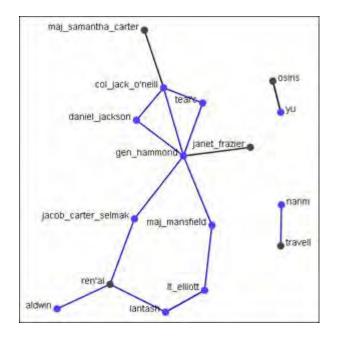
When the **agent-by-agent** network is opened all the nodes default to the color **red**. ORA has default colors for all the various NodeSets. But ORA can also color the nodes based on their individual attributes. Here's the default **agent-by-agent** NodeSet in red.

From the Visualizer main menu select **Actions > Color Nodes by Attribute or Measure**. This brings up the **Node Color Selector** dialog box.
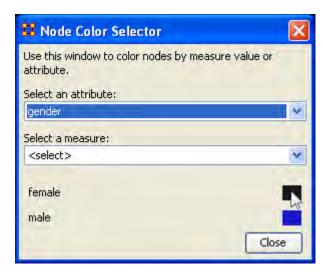
Selecting the **down arrow** for **Select an Attribute:**. Highlight an attribute from the list. ORA gives **default colors** to the nodes.
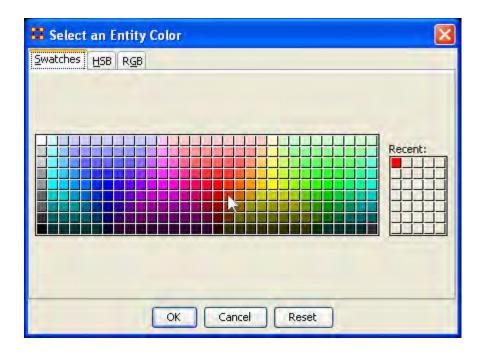
In some instances the color distinction isn't highly visible. And for presentation purposes a good contrast is necessary.

To bring up the **Color Selector** click on the color you want to change.

Note: *By* *hovering* *over the measure or attribute, you can bring up detailed tool tips.*
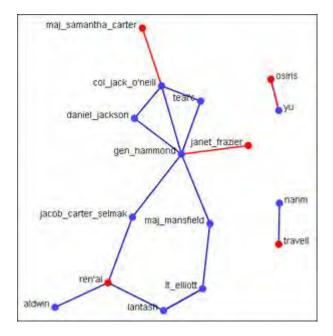
Select a new color from the swatches. Then select **[OK]**

Your new color choice is now reflected in the dialog box.

And the nodes with the attribute you selected are now a new, and brighter, color.

Compute Standard Measures

This functions runs four standard measures:

Cognitive Demand: Cognitive Demand measures the total amount of effort expended by each agent to do its tasks. Individuals who are high in cognitive demand value are emergent leaders. Removal of these individuals tends to be quite disruptive to networks.

Centrality, Betweenness: Betweenness tells us which node is the most connected to other parts of a network. For example, Betweenness can tell us which person in a network is the most central to the network as a whole. Betweenness measures the number of times that connections must pass through a single individual to be connected.

Centrality, Eigenvector : Eigenvector Centrality reflects one's connections to other well-connected people. A person connected to many isolated people in an organization will have a much lower score in this measure then those that are connected to people that have many connections themselves. This would be true even if such a WHO might have a high amount of actual connections (i.e. high degree centrality).

Centrality, Total Degree: Degree Centrality tells us the relative number of direct connections a WHO might have in a network; the higher the score the more likely a WHO might be likely to receive and potentially pass on critical information that flows through the organization.

These values can be viewed either by hovering over a node and revealing the tooltip or double clicking a node and bringing up the Node status dialog and selecting the Measures tab.

Isolates
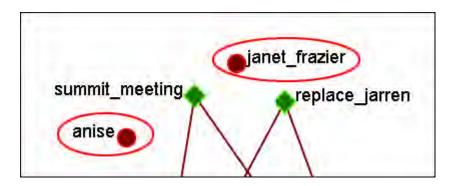
Periphery

Places isolates that were moved into column back to their original positions.

Hide Isolate Nodes
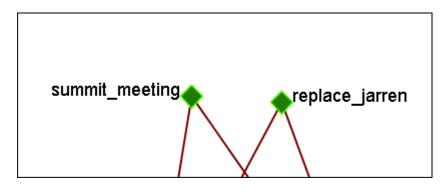
To further simply a visualization, it may prove useful to remove isolated nodes from the visualization.

Isolated nodes are not directly linked or connected to other nodes, which share direct links with each other. In the screen shot below, the red ellipses highlights an isolated nodes which need removed.

Select from the Main Menu: **Actions > Hide Isolated Nodes**.

In the screen shot below, after removing isolated nodes, the visualization is further simplified.

Combine into MetaNode

This option will combine all the Isolates into one MetaNode. This can be expanded right-clicking to bring up the contextual menu.
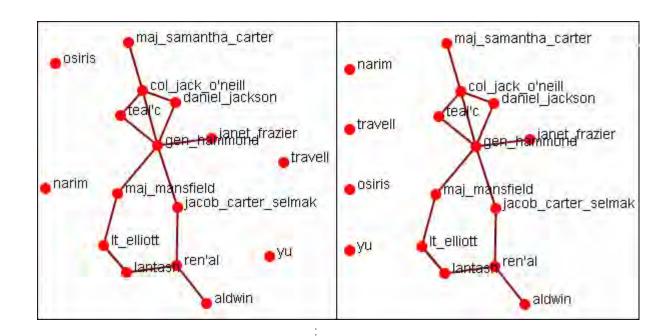
Move into Column

This function is used to move isolate nodes from the main viewing area to move easily view the connected nodes. These will be lined up on the left side of the screen.

NOTE : The original dataset was changed in order to create the isolates.

NOTE : If the names are not viewable and off the screen, change the position of the labels

The isolates can be returned to the original position using the Periphery options.

:

Add Network From Scratch

This function also allows the **creation of Networks from Scratch**. First, remove the check marks from all the NodeSets in the Legend to remove them from the screen.

The DataSet
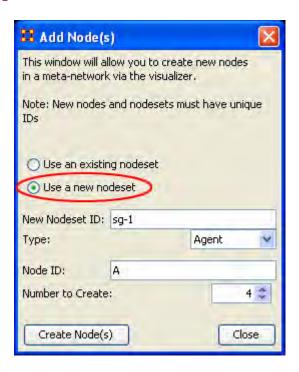
Next prepare your data for inclusion. Here is a small, five node, NodeSet.

A01	col_jack_o'neill
A02	maj-samantha_carter
A03	daniel_jackson
A04	teal'c
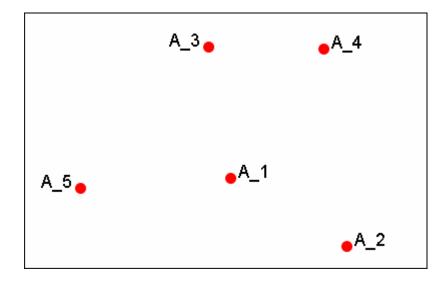
A05	gen_hammond

From the main menu select **Actions > Add Node(s)**.
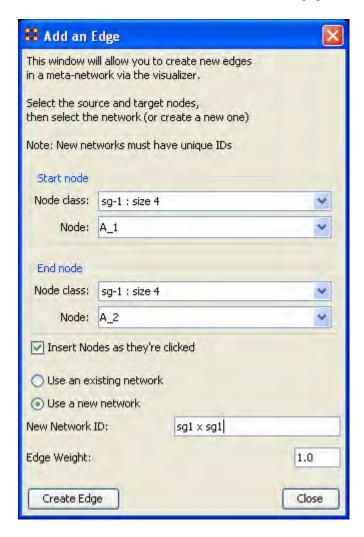
In the dialog box activate the **Use a new nodeset** radio button. Underneath, in the **Type:** dropdown menu, select the type of NodeSet. Then in the **New NodeSet ID:** textbox give it a name. Then select the **[Create Node(s)]** button.
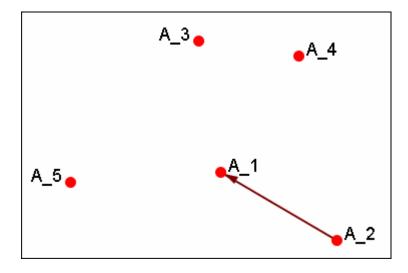
The five nodes will appear in the Visualizer with the prefix and attached numbers. Now it's time to connect the nodes.

From the Visualizer menu select **Actions > Add Link(s)**.
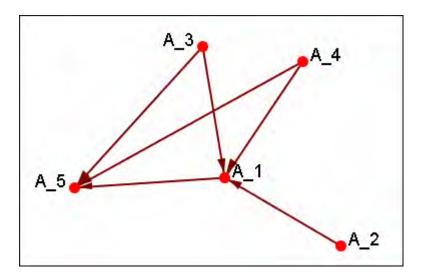
After selecting **[Create Link]** a new link will be drawn in the Visualizer.

Now switch the NodeSet to use to **Use an existing network**, select the newly created NodeSet in the dropdown, and create the rest of the links.

All of the links are now complete.

Save the Meta-Network with either **Save Meta-Network As...** or **Add Meta-Network to Ora** and use the ORA editor to add in titles and attributes.

Resize Nodes By Attribute
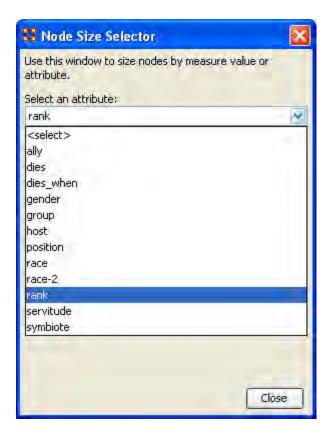
Resize by Attribute uses the numerical value of an attribute to size the nodes. Sizes will be scaled accordingly. Below is an attribute created for this example.

The numbers range from 0 to 10 and denote approximate rank. **10** denoting general/leader down through the ranks to **5** for civilians and finally **0** for persons not involved.

	Node ID	Node Title	rank
	<set filt... ⌄	<set filt... ⌄	<set filt... ⌄
☐	A01	col_jack_...	9
☐	A02	maj_sam...	7
☐	A03	daniel_jac...	5
☐	A04	teal'c	5
☐	A05	jacob_cart...	5
☐	A06	lt_elliott	6
☐	A07	ren'al	10
☐	A08	aldwin	5
☐	A09	janet_frazier	5
☐	A10	maj_man...	8
☐	A11	gen_ham...	10
☐	A12	lantash	7
☐	A13	narim	2
☐	A14	travell	10
☐	A15	osiris	0
☐	A16	yu	0
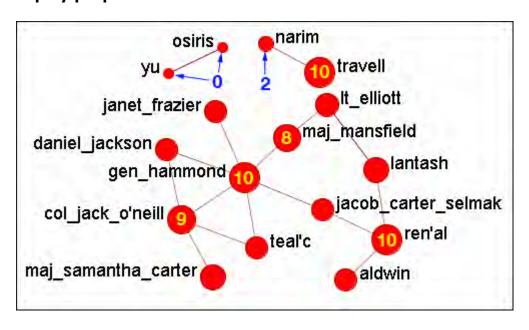
From the main menu select **Actions > Size Nodes by Attribute or Measure**. Then select the attribute to use for the sizing. When finished, select **[Close]**.

NOTE : Sizing nodes by attributes only works with attributes of numerical content. Alpha content will have no affect on the size of the nodes.

Each node will now be displayed according to the value in the chosen attribute.

NOTE : **The numerals are** **not** **displayed in ORA and are shown only for display purposes.**
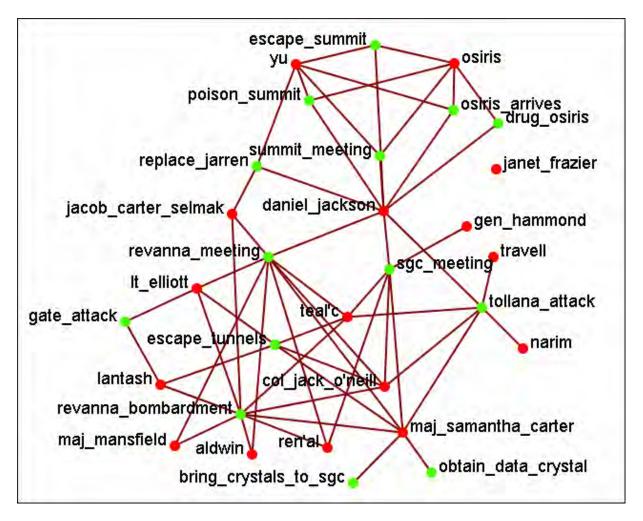
	Resize Nodes By Measure

ORA has the capability to visualize nodes by scaling them relative to each other in relation to common SNA Measures or Attributes as selected within the ORA Visualizer.
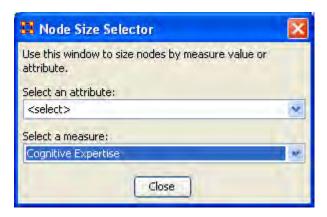
The key benefit to sizing nodes by Measure or Attribute is the ability to conceptually capture which nodes in your network figure more prominently in terms of measures by simply gauging their size visually and intuitively.

First, visualize one of your networks. Below is the Stargate **agent x event network** to find the agent with expertise regarding events in the scenario.
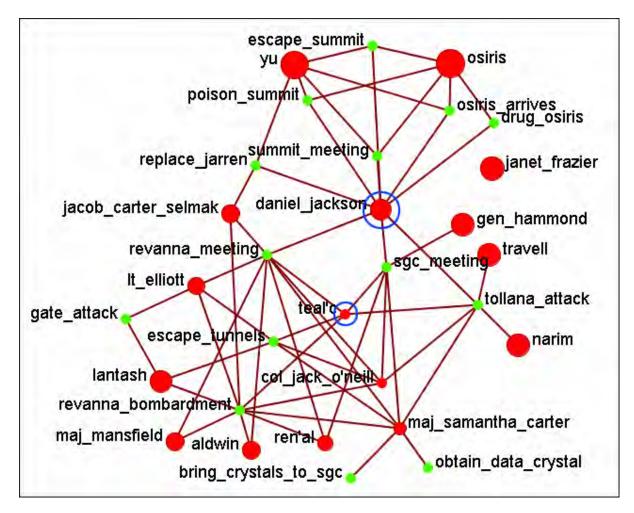
From the main menu select **Actions > Size Nodes by Attribute or Measure**.

From the bottom textbox I selected **Cognitive Expertise**. Each of the nodes in turn grows to reflect it's value from that measure.

The display now changes to reflect each node's value by size.

We'll compare two nodes: **daniel_jackson and teal'c**.

262

daniel_jackson's cognitive expertise value is 0.2400 and is displayed with a large red circle while teal'c's value is 0.1556 is displayed by a much smaller red circle. The values can be read in the **Node Status** dialog box.

Cognitive Measure for daniel_jackson

Cognitive Measure for teal'c

Visualizer Tools

The **ORA Visualizer** provides a suite of tools to visually analyze your Meta-Network. The Visualizer Tools can be located accessed from within Visualization mode on the main menu bar (image below):

Drill Down : With the ORA Drill-Down Wizard you can quickly visualize ego networks by overall node class **knowledge, tasks, resources, agents)** or by choosing individual nodes from a checklist. The ORA Drill-Down Wizard is only accessible through the Visualizer:

Node Status The Node Status Window is selected by default. It first pops-up when you select any node in the Visualizer. This window provides you with a snapshot of the currently selected node by displaying that nodes unique Attributes, Measures, and Neighbors.
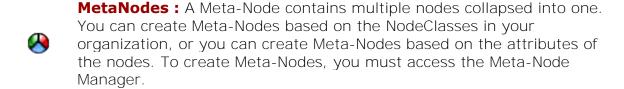
Group Viewer : Allows viewing of current display to be viewed in various grouping: **CONCOR, Newman, Johnson, FOG, and K-FOG**

Key Set Selector : Helps put a set of arbitrary items on the screen.

Legend : Allows for toggling visibility of nodes and links on the fly.

MetaNodes : A Meta-Node contains multiple nodes collapsed into one. You can create Meta-Nodes based on the NodeClasses in your organization, or you can create Meta-Nodes based on the attributes of the nodes. To create Meta-Nodes, you must access the Meta-Node Manager.

Path Finder : The Path Finder allows you to focus or **drill down** on a particular node, or multiple nodes, that you may be interested in analyzing in greater detail.

Sphere of Influence : Each node within a network has a unique **Sphere of Influence** or **Ego Network**, essentially it's direct relationship with it's neighbors as a function of specified path length.

Backward Influence : Used to view how a node influences other nodes.

Block Maps : Block Maps display varying intensities of data by displaying "blocks" correlating to the value of data. For instance, a block map displaying node centrality values would reserve the largest block for the node highest in that value. Color is used to as well to visualize different values with the the darkest color reserved for the highest value. Block Maps can also be referred to as heat maps.

Node Locator : Brings up a window with the Meta-Network in a collapsible tree structure. Selecting any node within this tree will highlight the node, enlarge it, and bring it to the center of the display.

Micro Simulation :

Meta-Network Designer :

Networks Over Time : Allows you to view a version of your network across multiple time periods.
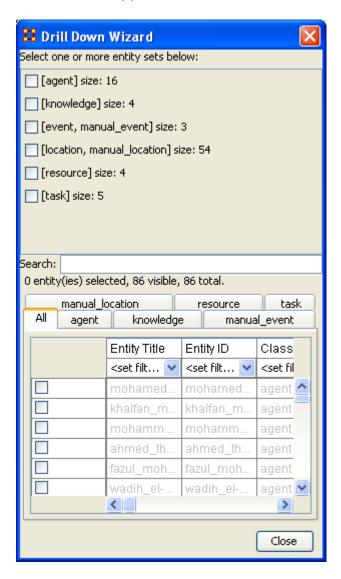
Vector Maps :

Drill Down Wizard

With the ORA Drill-Down Wizard you can quickly visualize **ego networks** by overall node class **(knowledge, tasks, resources, agents)** or by choosing individual nodes from a checklist. The ORA Drill-Down Wizard is only accessible through the Visualizer:

From the ORA Visualizer Main Menu select **Tools > Drill Down**.
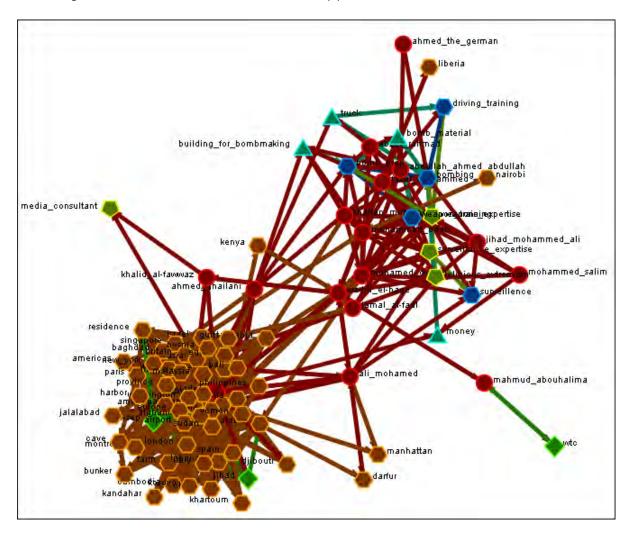
The following window box will appear.

The top sections allows you to chose the nodes for use. The bottom tabs allow for choosing specific nodes within a set.
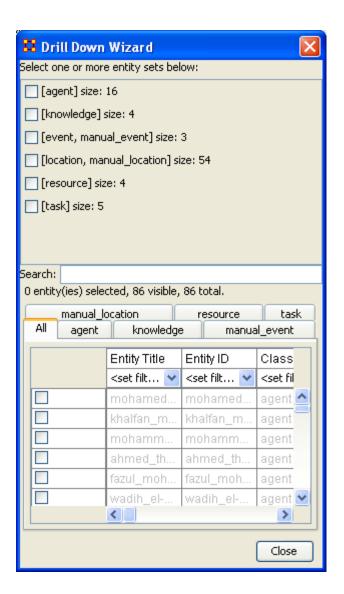
Drill Down Wizard Example

Using the Embassy Meta-Network, we will render the various node classes using the Drill-Down Wizard Tool. The screen shot below displays the Embassy Meta-Network as it should first appear in the Visualizer.
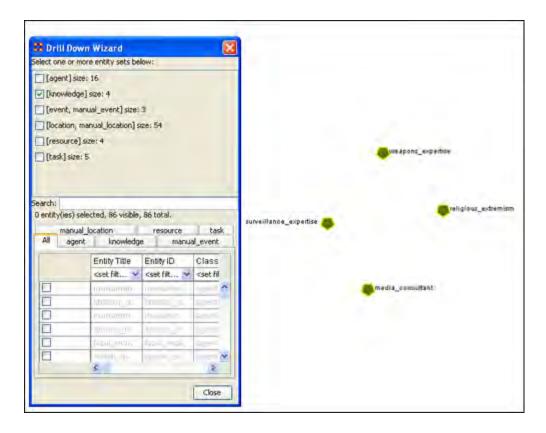
From the Main Menu select **Tools > Drill Down**.
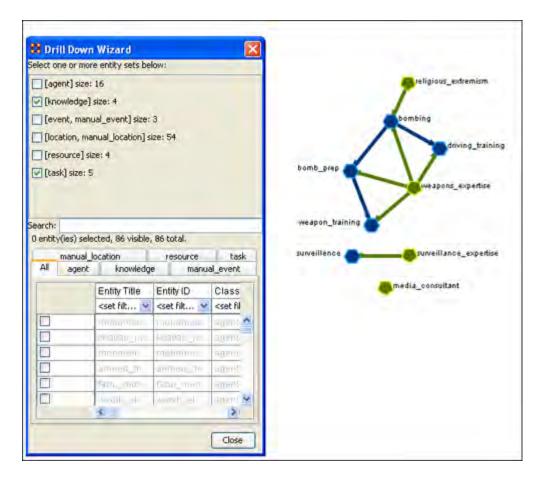
This will bring up the **Drill Down** dialog box as shown below).

Here, select the **node class Knowledge** by selecting the Knowledge check mark box. The screen shot below, displays the end result that you should now see in the Visualizer.
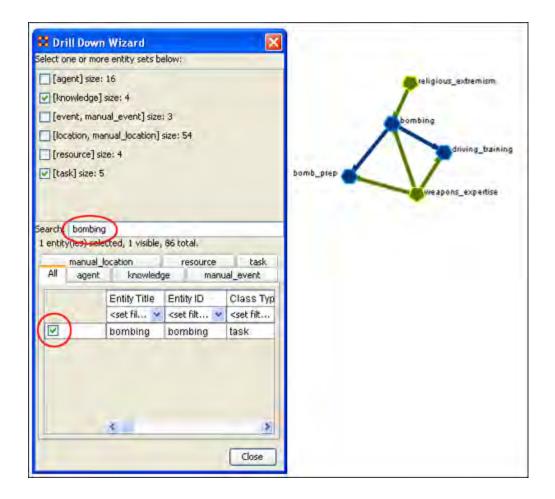
Next, with the Knowledge node box checked, add the **node class Task**.
The screen shot below displays the end result.

Next use the search bar feature of the Drill-Down Wizard to narrow the search.
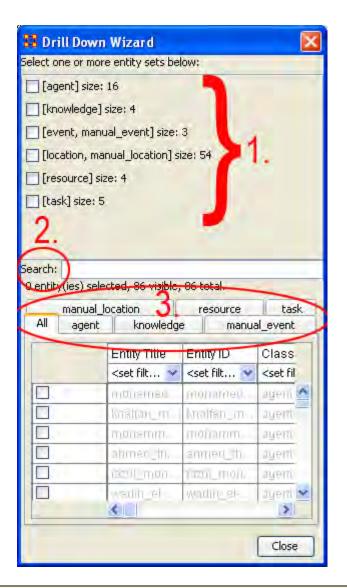
Enter the word **Bombing** in the Visualizer search field below. Then select the check box on the bombing row. The screen should now reflect your new search parameters. Everything not associated with **bombing** has been turned off.

Drill Down Wizard Explained
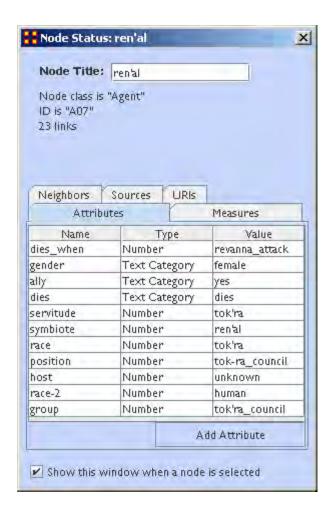
The ORA Drill Down Wizard can be broken down into three primary sections of input:

1. **Section 1:** enables you choose individual node sets or combination of node sets to display in the Visualizer.

2. **Section 2:** of the Drill Down Wizard enables you to search for a particular node within your Meta-Network. This can be handy when you dealing with large nodesets and the one you are interested in finding is not easily located.

3. **Section 3:** enables you **check mark** an individual node within your Meta-Network by node class if so desired.

Node Status

The **Node Status Window** is used to reveal pertinent information about one selected node. The window can be brought up by either selecting it from the main menu, **Tools > Node status** or by double-clicking the node in the visualizer. The dialog box below will appear.

The top of the box will contain general information about the selected node. Underneath there are five tabs available which will display the node's information. You can switch nodes by selecting any other node in the visualizer. ORA will automatically change the display.

Neighbors : displays the nodes which are linked to the selected node. It also displays the weight of each link.

Sources :

URIs : If there are any URIs attached to the node they are display under this tab.

Attributes : Types and Values of all attributes are displayed under this tab.

Measures : If Measures were run on the Meta-Network their values are displayed under this tab.

Group Viewer

The Group Viewer helps separate nodes into distinct groups.

We'll use one of the networks for each of the groups to further examine the breakdown.

There are six groupings which can be called up:

Clique :

CONCOR :

Johnson :

Newman : Used to find clusters in a network.

FOG :

Alpha-FOG :

ORA implements the newman algorithm presented in the following paper:

Finding community structure in very large networks, Aaron Clauset, M. E. J.
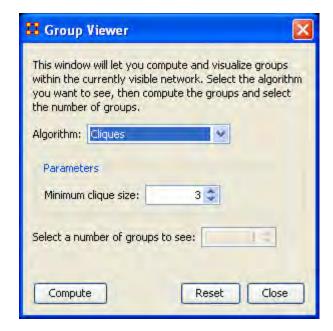
Newman, and Cristopher Moore, Phys. Rev. E 70, 066111 (2004). The algorithm uses link weights. It is not the Newman-Girvan algorithm based on betweenness centrality.

Please note, Newman developed this algorithm as it scales better than the Newman-Girvan for large networks.

Group Viewer-Cliques

Below is an image of the Group Viewer box and procedures for using this tool.
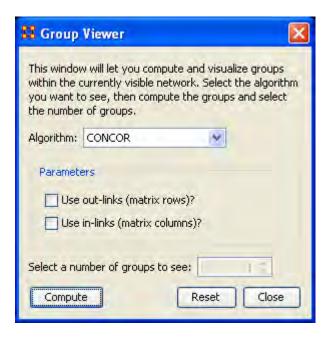
Parameters

Minimum clique size :

🔲 Group Viewer-CONCOR

Below is an image of the Group Viewer window and procedures for using this tool. This is a Structural Equivalence/Clique which can be used on one or two mode data, and one or a set of networks at once.

274

Parameters
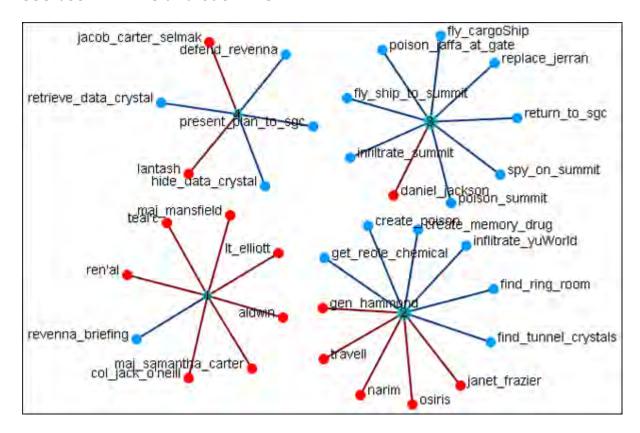
Use out-links (matrix rows)? The links leading **out of** the chosen node into another node.

Use in-links (matrix columns)? The links leading **into** the chosen node from another node.
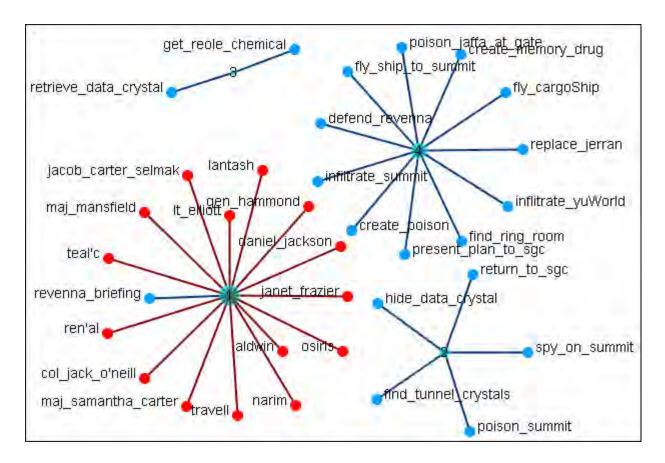
Example

The example uses the network **agent x task**. Foe each of the three possibilities *in, out, both* the display is shown with the number of groups set to **4**. The resulting displays show how the in and out links affect the groupings.
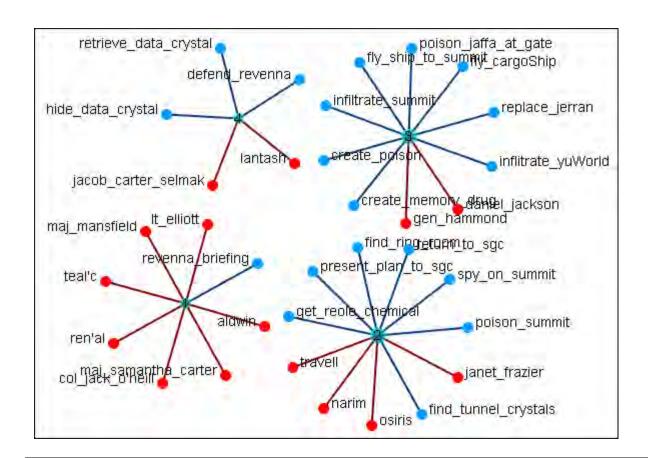
Use both in-links and out-links

Use in-links

Use out-links

276
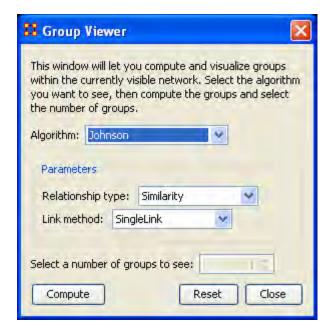
	# Group Viewer-Johnson

Below is an image of the Group Viewer Johnson window and procedures for applying the Johnson algorithm. This is a Hierachical Clustering based on Similarity Standard hierarchical approach that builds groups based on similarity scores.

Parameters

Relationship type:

- Similarity
- Dissimilarity
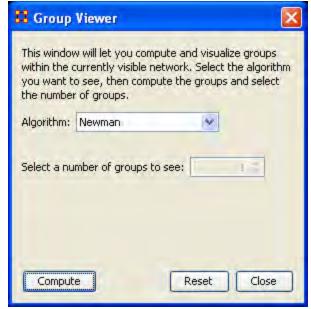
Link method:

- SingleLink
- CompleteLink
- AverageLink
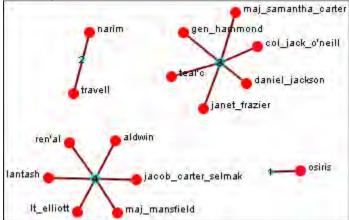
 # Group Viewer-Newman

- Below is an image of the Group Viewer window and procedures for applying the Newman grouping algorithm:

- The Newman grouping is used to find clusters in a network. I began increasing the number of groups to view until the larger cluster began to break up.
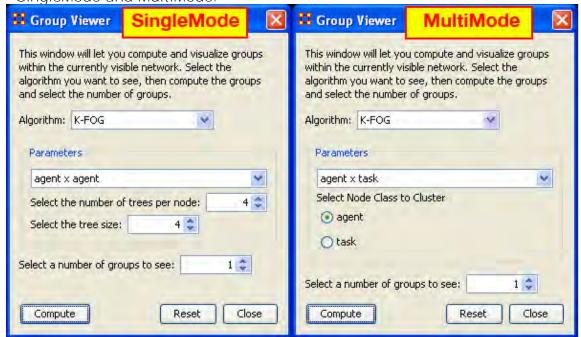
- In this case the nodes remained clustered (with only one or two nodes breaking off) until I reached **4 as the number of groups**. Then the network broke up into two large and two small clusters. The nodes in each of the two large clusters have a commonality as to the organizations they belong to and the tasks they were assigned.

🔲 Group Viewer - K-FOG

- **FOG (or Fuzzy Overlapping Groups)** works in either square-mode (NodeSet-1 x NodeSet-1) of multi-mode (NodeSet-1 x NodeSet-2). The multi-mode is the more natural procedure. it's prime function is to form nodes into groups (clusters or communities) and show which nodes share the secondary functions.

- **REMEMBER :** Groups produce tasks - people participate in tasks.
- Below are images of the Group Viewer K-Fog window for both SingleMode and MultiMode.
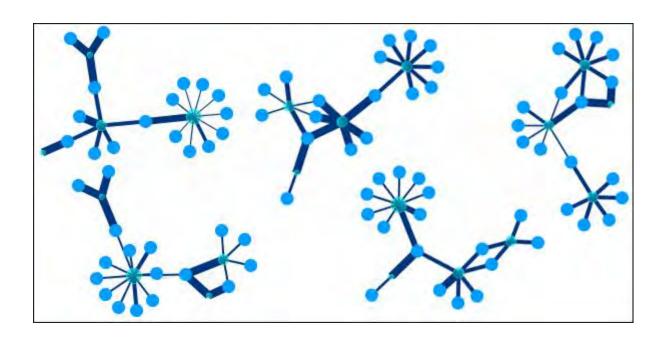
-
- **Parameters**
- **SingleMode and MultiMode**
- **Select a Network:** The same in both. The dropdown menu lists all the available networks in the selected Meta-Network.
- **Select a number of groups to see :** Number of groups to divide the nodes into. Generally the analyst would have an idea of the number of groups that are necessary.
- **SingleMode**
- **Select the number of trees per node :** The number of sample events to generate.
- **Select the tree size :** The number of people involved in each group
- **MultiMode**
- **Select Node Class to Cluster :** Selects, via the radio buttons, the node class to display using the other node class as it's base.
- **Links**
- The thickest link is usually the leader of the group. The question to ask is **WHY** is that particular person the strongest link. It is important to look at the composition of the group(s) involved.
- If the weights are Lopsided (many strong and/or weak links) the analyst needs to look at the association of the **interstitial*** actors whose information will assist in analyzing the groups.
- If a node is connected to only one group and that group has few actors then it is considered somewhat **insular**.
- **SingleMode Example**
- Run on the **agent x agent** network with a number of threes per node set to "2", the tree size set to "2", and the number of groups to see

set to "2". The header designation is "S=the set (1-5)" and "G=first or second group".

	S1-G1	S1-G2	S2-G1	S2-G2	S3-G1	S3-G2	S4-G1	S4-G2	S5-G1	S5-G2
col_jack_o'neill	0.2	0.125			0.1667	0.1667	0.2381	0.2		
gen_hammond	0.4375	0.5	0.375	0.5	0.1667	0.5	0.2857	0.6	0.5789	0.1176
osiris			0.125	0.1	0.25	0.0416	0.0952	0.1334		
yu					0.3334	0.0416				
lantash					0.25	0.0416			0.0588	0.1052
daniel_jackson					0.0833	0.0416				
travel							0.0476	0.1334		
anubis							0.0476	0.2667	0.1579	0.1176
ren'al							0.0476	0.2667		
maj_samantha_carter									0.1579	0.0588
maj_mansfield									0.0526	0.0588

- **MultiMode Example**
- The **agent x task** network is used and the **Task Node Class** is selected to Cluster. THe number of groups to see was set to **"4"**. The results of hitting **[Compute]** five times are shown below. Each of the results is different but there is an overall pattern. In four of them there is a distinctive single large cluster of nodes.
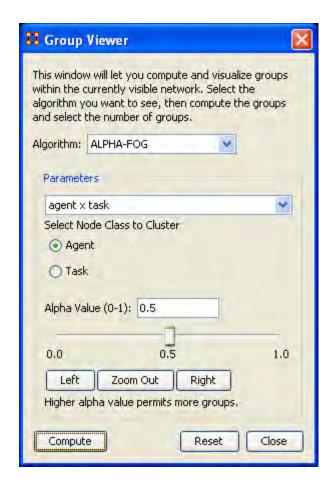
Group Viewer-ALPHA FOG

Alpha-FOG is similar to K-FOG in it clusters nodes into groups. The difference between the two is Alpha-FOG is used when the analyst is not quite sure of the number of groups.

The trade-off between K-FOG and Alpha-FOG is accuracy vs. insight.

The ALPHA-FOG dialog box

Parameters

The dropdown menu contains all the networks in the Meta-Network. After selecting a network ORA will display the two nodesets in the network below in a set of radio buttons. Select one to display. The other will be used as the tasks set.

Alpha Value (0-1):
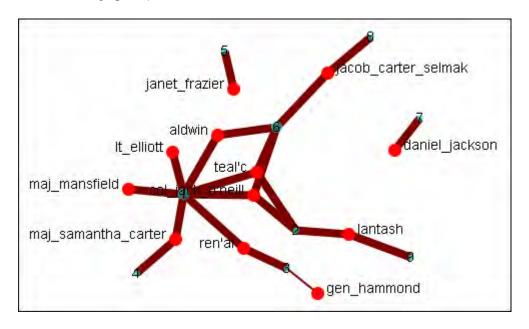
The slider bar controls the ALPHA-value. It defaults to 0.5. You can manually move the slider right or left then press **[Compute]**.

Below the slider are three buttons, **[Left], [Zoom Out], and [Right]**.
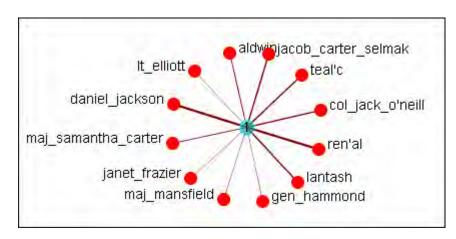
- Pressing the **Left button** will reset the three values. The left value will remain the same, the right values will be reduced by half, and the middle value will be the value somewhere between the two. Each press of the Left button will repeat the sequence. Pressing the Right button will do the same, but in the opposite direction.

- The **Zoom Out** button reverse the above procedure in the exact order **(i.e. if you pressed Left, Right Right, Left, and Left the Zoom Out button would undo then as Left, Left, Right, Right, Left)**.

The buttons at the bottom control the display. The **[Compute]** button uses the current Alpha value and re-display the network. The **[Reset** button removes all groups and resets all the nodes to default. The **[Close]** button closes the dialog box.
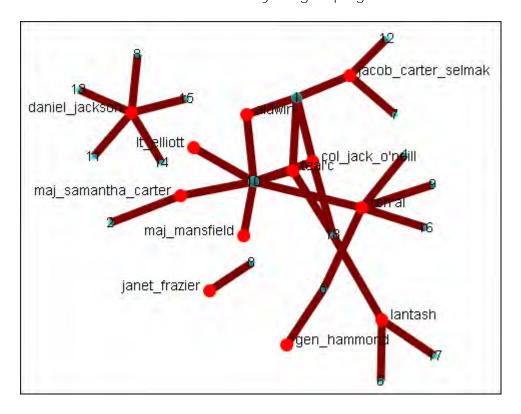
Below is the initial display using the default of **0.5**. It doesn't really tell us much about the grouping of the network. There are some groupings but there are also four groups with one node only and many small "almost" groupings. Also all the links are the same weight which does not indicate the leader of any group.

Setting the slider bar to **0.0** groups all the nodes into a single group. This contains the opposite problem as there is only one group.
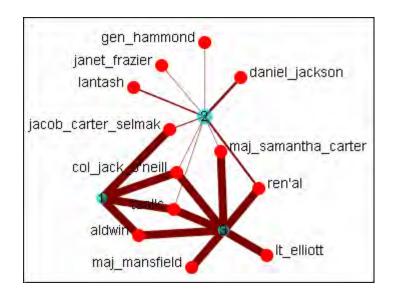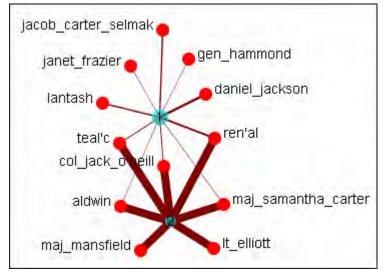
Setting the slider bar to **1.0** divides the tasks into **18** groups, none of which are distinctive. There is definitely no groupings here.

It's necessary to find the optimum setting.

Starting from the default setting I clicked Left. The range was from **0.0 – 0.5**. got three groups which looked fairly good. But I decided to check out the other options. I clicked Left again and the range changed to **0.0 – 0.25**. But it combined all the nodes into one group. Definitely was he correct grouping. I clicked Right and the range changed to **0.125 – 0.25**. It split them into two groups.
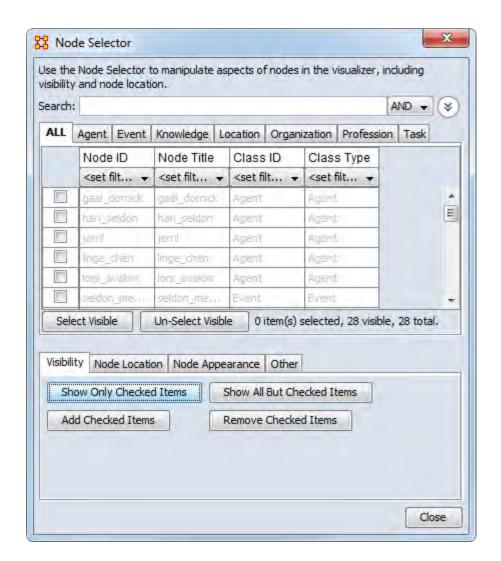
Node Selector in Visualizer Tool

You can use the Node Selector to simply complex visualizations. The Node Selector tool is accessible from the Visualizer menu: **Tools > Node Selecter**.

Node Selector

Search : Displays only nodes which have matching content. To the right is the **AND/OR** dropdown box where you can select whether to make ORA search for entries with all terms in the box or terms in the box.

NodeClass tabs : Contains a tab for each NodeClass in a Meta-Network.

Node Display Area Displays all the nodes for the currently active tab. The checkbox to the left allows you to select/deselect nodes to work with.

Select Visible/Un-Select Visible buttons : This works on all nodes currently displayed in the Node Display area. If places a checkmark [selects] or removes a checkmark [deselects] on al items visible.

Visibility

Show Only Checked Items : Shows only nodes **which have** a checkmark next to their names.

Show All But Checked Items : Shows only nodes **without** a checkmark next to their names.

Add Checked Items : Adds to the visualizer display (if they are not already visible) any nodes which have a checkmark to the left of their names.

Remove Checked Items : Hides in the visualizer display (if they are not already hiddne) any nodes which have a checkmark next to their names.

Node Location

Move to the Left : All selected nodes will be moved to a column to the left side of the screen.

Move to the Right : All selected nodes will be moved to a column to the right side of the screen.

Node Appearance

Show Label(s) : Works in co-ordination with the [**Set Labels**] button and all selected nodes. Will make the lables visible [checkmarked] or hidden [no checkmark].

Set Labels button : Clicking this button activates your choices with the **Show Label(s)** option.

Node Color : Brings up a color palette to chose a color.

Change Color button : The color of any node(s) selected will be changed to the new **Node Color**.

Other
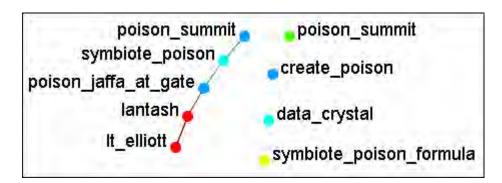
Create Meta-Node : Creates a Meta-Node from all selected nodes.

Highlight Checked Items : Draws an outline around all nodes selected. This functions with nodes of all sizes.

Node Selector Examples
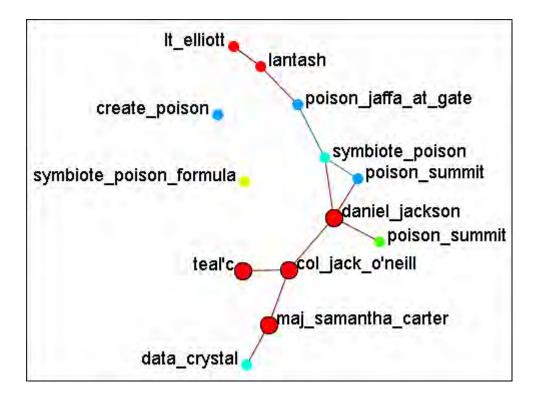
Following is a short example of the KeySet Selector. It covers the main aspects of the tool only.

In the **Filter Commands** textbox the word **poison** is typed which causes all nodes with **poison** in any attribute to be revealed. Pressing the **[Select All]** button places a check mark in all the nodes. Lastly in the **Visualizer Commands** pressing the **[Show Only Checked Items]** button hides all but the checked items.
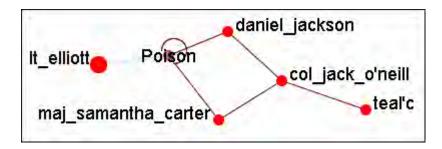
The **[Clear All]** button is pressed to clear all the check marks. **poison** is replaced in the **Filter Commands** with the word **sg-1**. Be aware that even though only nodes with **sg-1** are to be added the two nodes with **sg-17** also appear as sg-1 is a sub-string of sg-17. SO instead of using the **[Select All]** button check marks are manually placed in the check boxes desired.

The **[Add Checked Items]** button is pressed and the four new nodes are added to the visualizer.
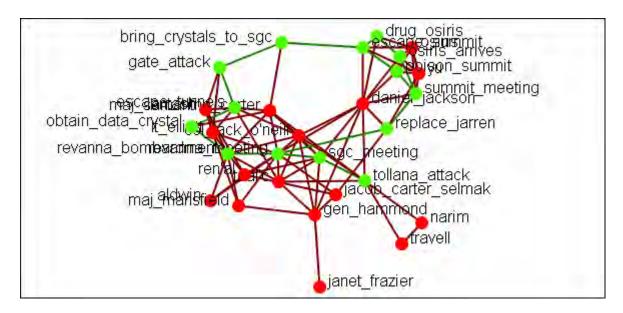
With the four nodes of sg-1 still check marked press the **[Highlight Checked Items]**. The four nodes are now displayed in a larger format.

The last button, **[Create Meta-Node]** will cull nodes into a Meta-Node.

Move Left/Right
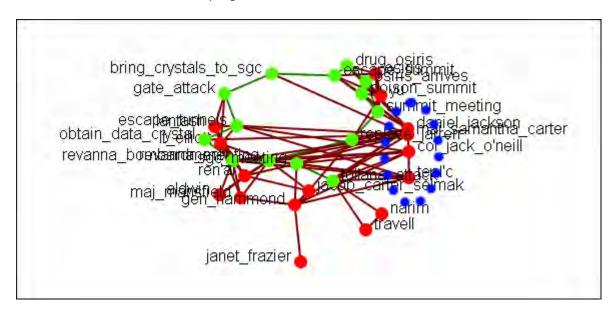
If a set of nodes needs "pulled out" of the display the **Move Left/Move Right** buttons can help. Below is a normal display. The nodes for SG-1 are quite difficult to see.
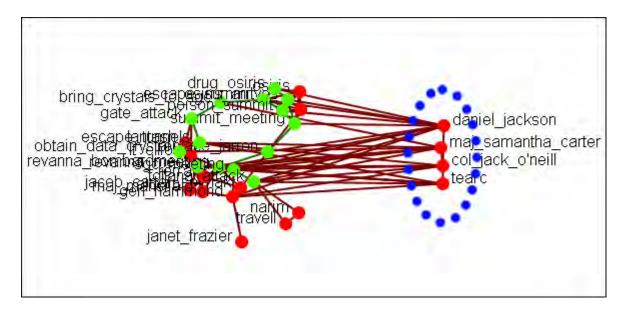
From the Node Selector dialog select the nodes to be highlighted and place a check mark in the box.

Press the **[Move to the Right]** button. This brings the selected items toward that side of the display.
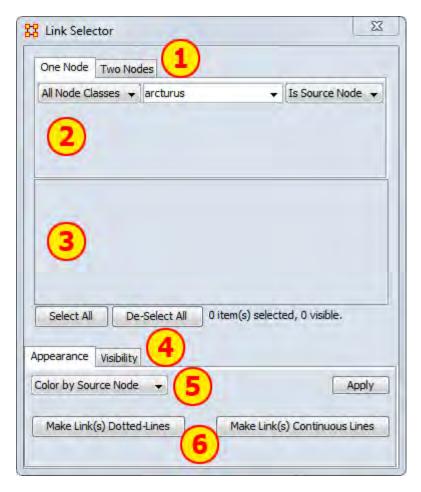
Pressing the button again will move the selected nodes farther in that direction.

Link Selector

In a complex network you may find it difficult to locate one particular link. This is where the **Link Selector** will help.You can chose to work with **One Node** or **Two Nodes**, picking which is the **Source/Target** node, and the appearance of the link in the Visualizer.
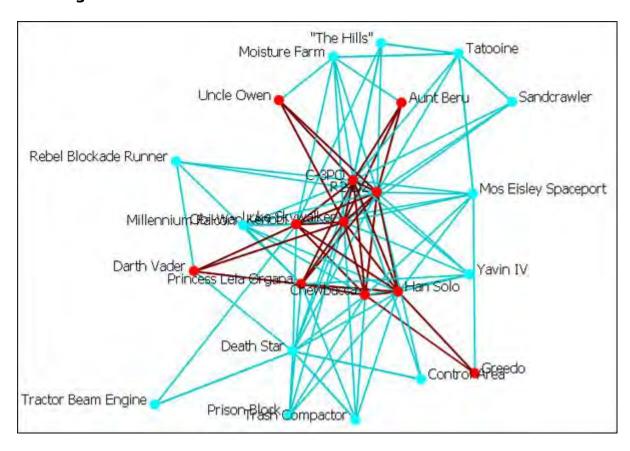
1. **One or Two Nodes :** Chose to specify links using methods for One Node (shows all nodes linked to node picked) or Two Nodes (where you specify the exact two links to use) or

2. **Node Area :** In **One Node** you pick the node and ORA shows you all the links. In **Two Nodes** you pick two nodes the chose one of three **direction options :** From First to Second, From Second to First, Goes Either Direction. This last one is handy in order to change the apprearance of bi-directional links.

3. **Link Area :** Displays all links that meet the parameters in the Node Area. Beneath this are the [**Select All**] and [**De-Select All**] buttons.

4. **Appearance/Visibility :** This allows you to control the look of the links. **Appearance** allows you to change link color and make then dotted or continuous lines. **Visibility** allows you to **Hide or Show** links.

5. **Link Color :** The drop down menu has three options : Color by Source node, Color by Target Node, or Choose a Custom Color.

6. **Dotted/Continuous Lines :** Allows you to choose the style of line.

The Original Network

17 APR 11
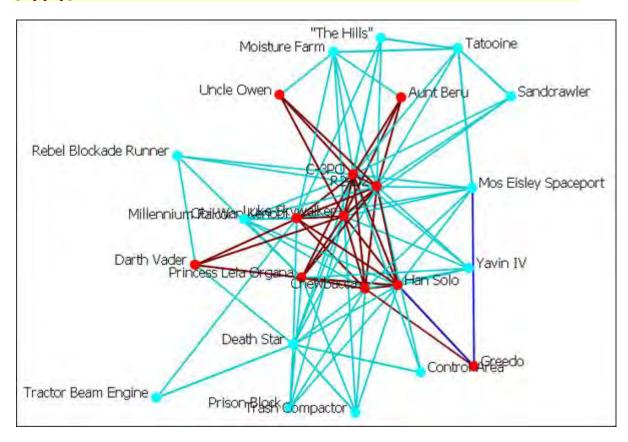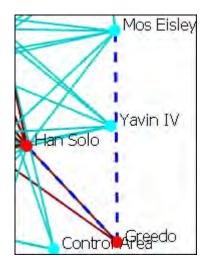
Link Selector-One Node

In [Area 1] Select the [**One Node**] tab. In [Area 2] select a node from the drop down menu. To the right select from the other drop down menu

whether you want this to be a **source** or a **target** node. All links associated with this node will be displayed in [Area 3]. Click on links you want highlighted. Click the [**Appearance**] tab in [Area 4]. Select [**Choose a Custon Color**] from the drop down menu in [Area 5]. In the Color Select dialog box select a complementary color and click [**OK**] then click [**Apply**].

Example : I select the node **Greedo** and place checkmark in both boxes. Under the tab **Appearance** I select [**Choose a Custom Color**] from the drop down menu. A distinctive dark blue color is selected then I click [**Apply**].

You can further enhance this information by use of the [**Make Link(s) Dotted-Lines**] button.

Notice that in this example the link **Greebo to Mos Eisley** is a clearly defined dotted-line. The link **Greebo to Han Solo** is a blue dotted line but the link **Han Solo to Greebo** is still the default red and could obscure it in a more densely packed network.

Caution should be observed when working with a large network.
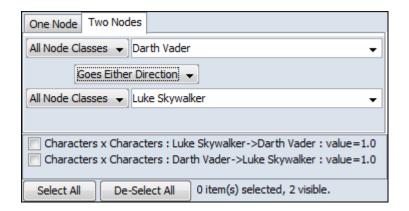
When finished you can, from the main menu, select **Display > Link Appearance > Link Color** and chose either **Match Source Node** or **Match Target Node** to reset your colors. Another choice is to select the particular links and change the colors via the method described above. You can also change any dotted-lines back using the [**Make Link(s) Continuous LInes**] button.

17 APR 11
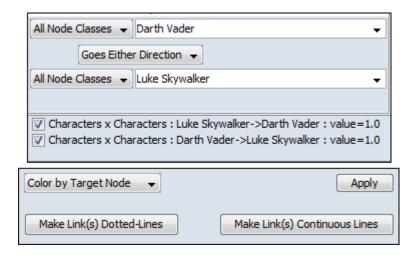
Link Selector-Two Nodes

Two Nodes
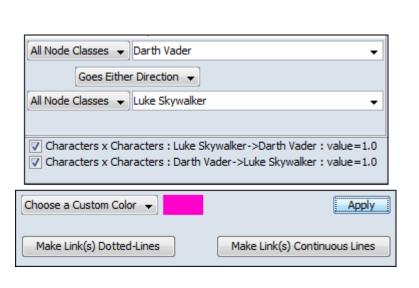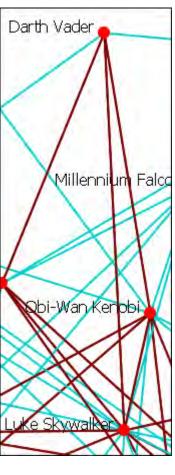
In [Area 1] Select the [**Two Nodes**] tab. In [Area 2] select a node from both of the drop down menus. Between these menus is another drop down menu about whether you want this to be a **From First to Second**, **From Second to First**, or a **Goes Either Direction**. Choosing either of the first two will result in a single link. Choosing **Goes Either Direction** will give

295
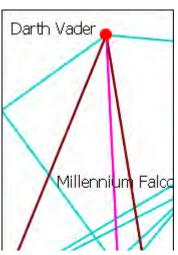
you one link (if the link is bi-directional) or two links (if the link is bi-directional). The links will be displayed in [Area 3]. Click on links you want highlighted. Click the [**Appearance**] tab in [Area 4]. Select [**Choose a Custon Color**] from the drop down menu in [Area 5]. In the Color Select dialog box select a complementary color and click [**OK**] then click [**Apply**].

Two nodes have been selected, **Darth Vader** and **Luke Skywalker**, with the option [**Goes Either Direction**]. A check mark is placed next to both links. At present both links are their default color, red.
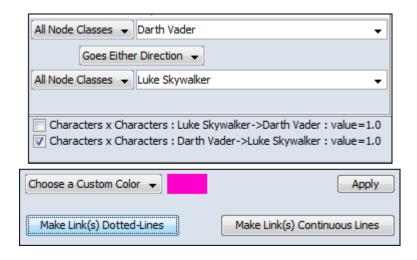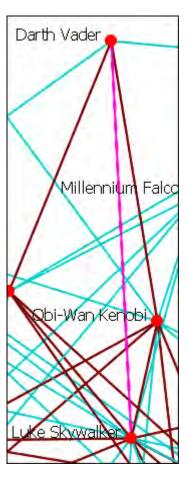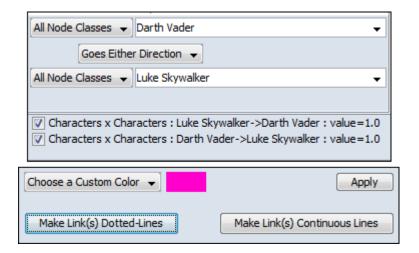
Under the appearance tab select from the drop doen menu [**Choose a Custom Color**]. Click the color block to the right and select a complementary color. For example, **Fuchsia**. CLick the [**Apply**] button and the links then take on that color.
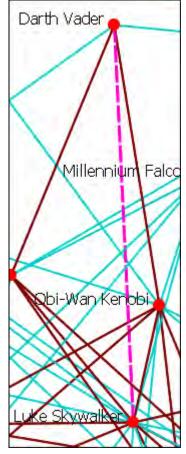
You can see how well they stand out now.

Notice here that a check mark is placed only next to **Luke Skywalker**. When the [**Make Link(s) Dotted Lines**] is clicked the dotted line is obscured by the continuous line from **Darth Vader** to **Luke Skywalker**.

Making them both Dotted Lines helps clarify the connection.

17 APR 11

Legend

The **Legend** dialog box allows you to quickly hide and reveal nodes and links in the visualizer.
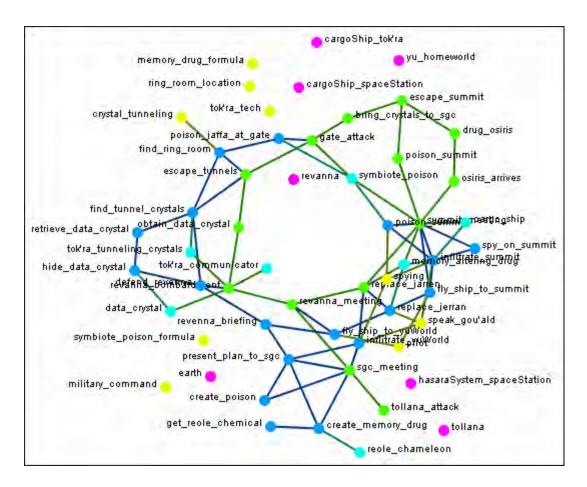
Use the check boxes to change a NodeSets visibility. When you first call up the Visualizer all the boxes will contain checkboxes…

…and in the Visualizer all nodes and links are revealed. A little hard to make heads or tails of something like that.
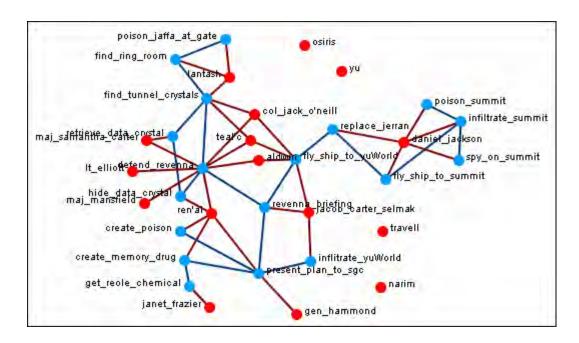
By using the Legend check boxes you can selectively **show/hide nodes and links**. Deselect the checkbox to the left of **Agent**. Instantly all of the Agent nodes are hidden along with any connections they had with the rest of the nodes. The node structure is a little easier to see now.

Next think about a particular set of connexions you'd like to see, say a **task x task network and how the agents relate to the tasks**.
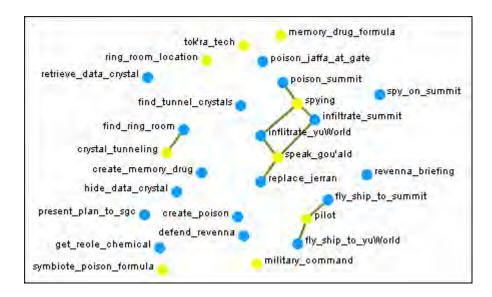
From the Legend box menu select **Control > Hide All nodes** Then again from the menu **Control > Refresh Legend**. Now place check marks in the check boxes for **agent, task, and task x task**. All the tasks will be linked together as per your network and all the agents will have connexions to the tasks they are associated with.
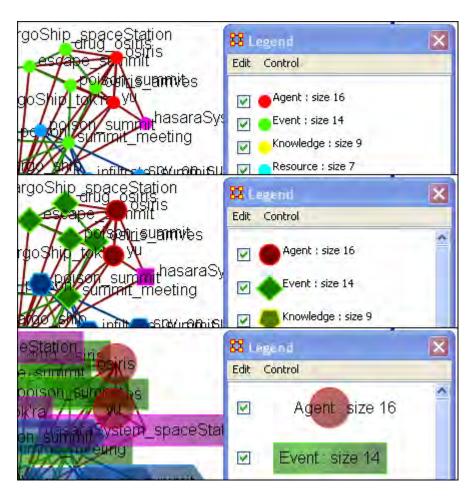
Another example is how **knowledge relates to specific tasks**. From the Legend menu select **Control > Hide All Nodes**. Now place check marks next to **knowledge, task, and the Knowledge x task network**.

Now the Visualizer reveals which knowledge is required for which specific tasks.

The legend also reflects the size, shape, and color of the nodes in the Visualizer.

Legend Edit Menu

Below is an image of the Legend Edit Menu and a description of this tool:

Edit	Control
Copy Legend Image to Clipboard	

Copy Legend Image to Clipboard : Copies an image of the **entire** legend that can be pasted into an image-editing program.

Legend Control Menu

Below is an image of the Legend Control window:

Control	
Show All Entities	
Hide All Entities	
Refresh Legend	

Show All Nodes : Will bring into view all entities in the Meta-Network. Good to use if you need to make a fresh start.

Hide All Nodes: Hides every node. Good starting point when you only wish to display a small subset of nodes.

Refresh Legend : After using the menu items above and/or using the check boxes this item will bring the visualizer up to date.
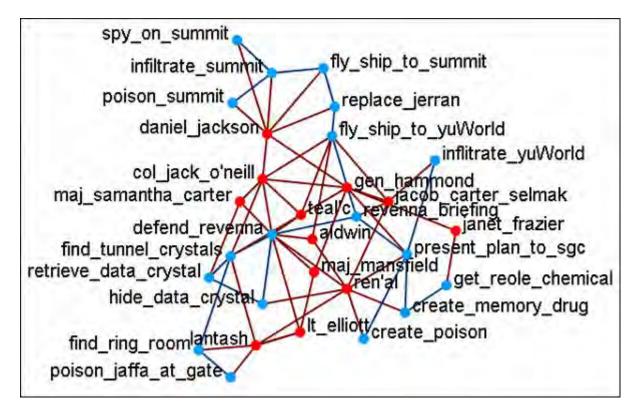
Node Color
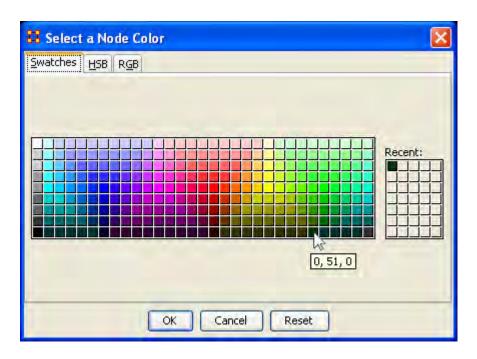
The color of a NodeClass can be changed in the Visualizer. In the Legend window click on the colored node of a NodeClass. This will bring up a color selector box. Select a new color and click **[OK]**. All the nodes of that NodeClass will change colors.

NOTE : This color change is in effect for as long as the Visualizer is open. If you close and re-open the Visualizer the colors will revert to their default colors.
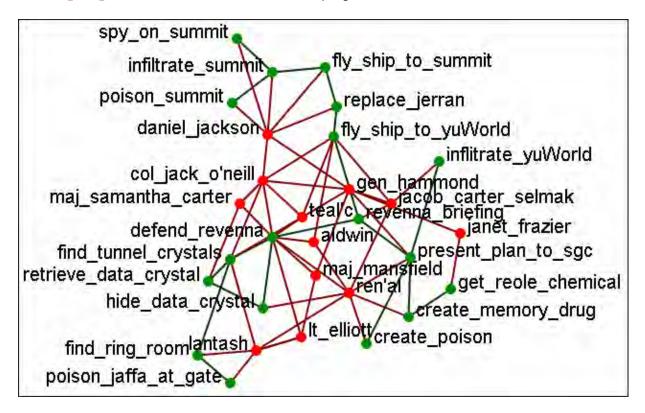
Here is a section with the default colors of **red** for agents and **cyan** for tasks.

Open the Legend window (if it's not open already) and click once on the NodeSet you want to change. This will bring up the **Color Change** dialog window.

Select **[OK]** and the new color will be displayed for the chosen NodeSet.
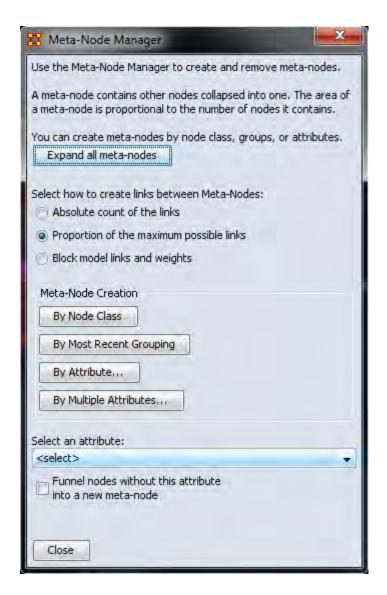
The newly selected color appears in both the Visualizer and Legend windows.

Create a Meta-Node

A **Meta-Node** contains multiple nodes collapsed into one. You can create Meta-nodes based on the nodesets in your organization, or you can create MetaNodes based on the attributes of the nodes. From the drop down menu select **Tools > Meta-Nodes** or select the **Meta-Node Manager** button on the toolbar. This brings up the **Meta-Node dialog box**.

Expand all meta-nodes : After any Meta-Nodes have been created this button will expand all Meta-Nodes out to individual nodes.

Select how to create links between Meta-Nodes:

Absolute count of the links : Counts any link from any node in the Meta-Nodes to any other node as 1.0. Uses this as the total

Proportion of the maximum possible links : Takes the total number of links in the Meta-Nodes and divides this by the totasl possible number of links.

Block model links and weights : Uses two terms. A) The Proportion of the maximum possible links (as calculated above); and B) The total number of links in a Meta-Network divided by the total number of possible links in a Meta-Network. If (A) is higher

than (B) then the value used equals (A). If (A) is lower then the value is 0.

Meta-Node Creation is controlled by four buttons:

> **By Node Class :** Creates Meta-Nodes based on current NodeSets. All nodes from a nNodeClass will be collected into a single node.

> **By Most Recent Grouping :** Creates Meta-Nodes based on the most recent use of the **Group Viewer**.

> **By Attribute... :** Toggles the **Select an attribute** pane below. Using the **Select an attribute** dropdown below you can create a Meta-Node group based on one attribute.
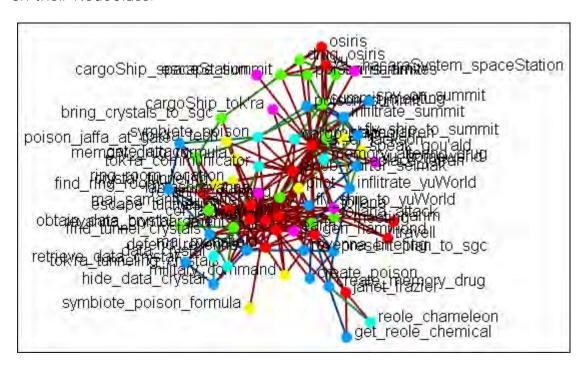
> **By Multiple Attributes... :** Allows for a more precise creation of Meta-Nodes based on drilling down through attributes.

Funnel nodes without attribute into a new meta-node : Groups all nodes which do not match the parameters and places them into one common meta-node.
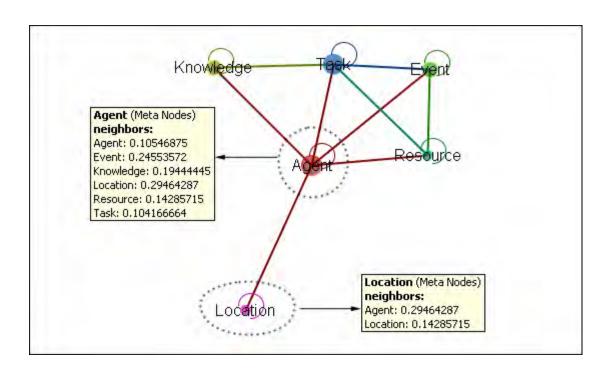
Meta-Node - Node Class

Using the **By Node Class** button separate all nodes into Meta-Nodes based on their NodeClass.
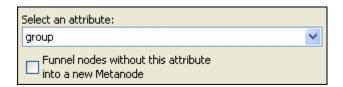
This button will create Meta-Nodes for each NodeSet.
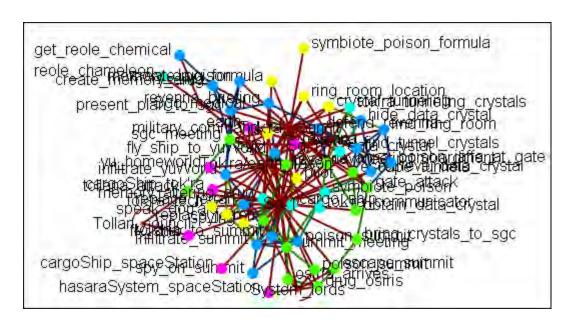
Meta-Node Attribute
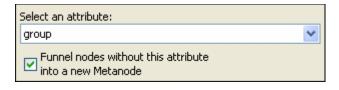
This button creates Meta-Nodes using a **single** attribute. It's used in conjunction with the **Funnel Nodes** checkbox. Below is an image of the Select an attribute box:
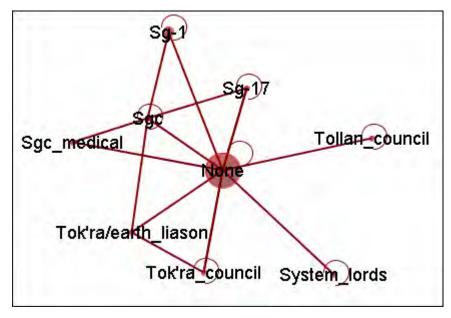
Choosing **group** without the **Funnel Nodes** checked creates a confusing display.
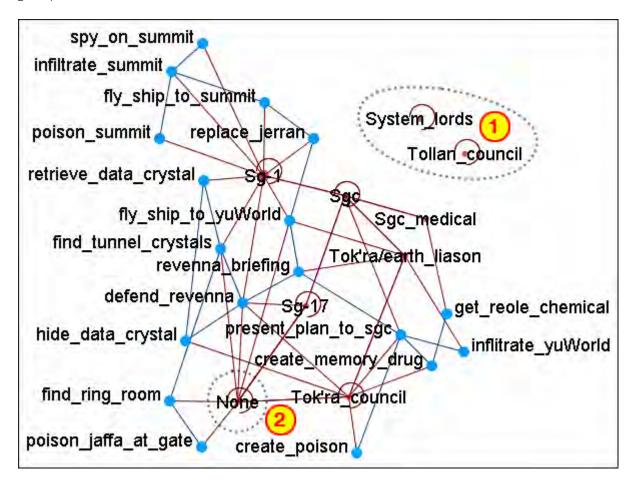
Next place a checkmark in the **Funnel nodes** box and select **groups** for a cleaner display.
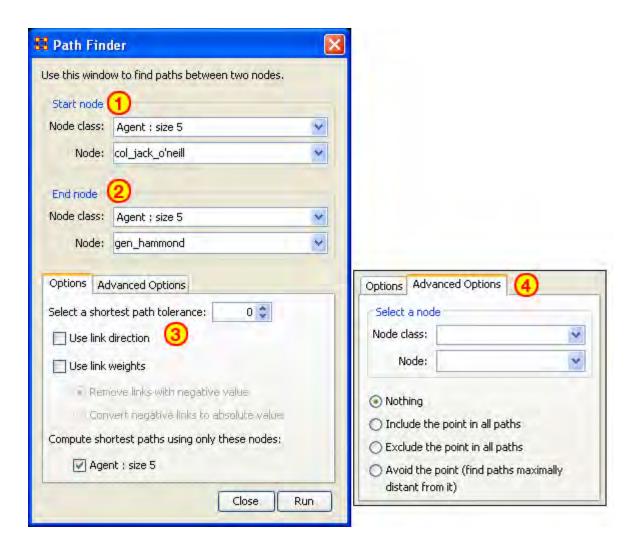
Leaving the **Funnel nodes** option unchecked works best with fewer NodeSets. Below only the **agent and tasks** NodeSets were used. Without funneling the individual tasks can be viewed as to which agent Meta-Node they connect to. It also reveals that groups in area **1** are not connected to

309

any tasks and that groups in area **2** are all the agent nodes that had no groups affiliation.

![Path Finder icon] **Path Finder**

The Path Finder is used to find, not only, the shortest path between two nodes but also paths to other nodes when particular situations occur. Below is an image of the The Path Finder tool and a description of its various elements.

The interface contains four sections:

Start node : Select the NodeSet/Node to begin the path.

End node : Select the NodeSet/Node to end the path.

Options : Options regarding link parameters and which NodeSets to use.

Advanced Options : Options regarding whether, and how, to use one particular node within the set.

Using only Agent nodes (tolerance = 0)
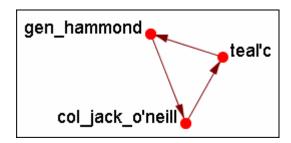
Using only the Agent NodeSet the pathway requires three intervening nodes.

	col_jack_o'neill	maj_samant...	daniel_jackson	teal'c	gen_hammond
col_jack_o'neill	0.0	0.0	1.0	5.0	0.0
maj_samant...	1.0	0.0	1.0	0.0	0.0
daniel_jackson	-1.0	0.0	0.0	-1.0	0.0
teal'c	0.0	0.0	0.0	0.0	1.0
gen_hammond	-5.0	0.0	0.0	0.0	0.0

NOTE : For this example the values in the Network have been altered from the regular values.

Tolerance of "1", nothing else checked.

Shortest path: 1
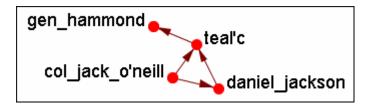
*There is the direct link from **gen_hammond to col_jack_o'neill** and the secondary pathway with **teal'c** in the middle.*

Use link direction

Shortest path: 2

The path direction **must** lead **from** the source **to** the target.
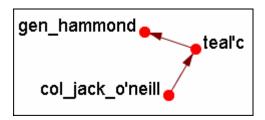
*The link **from gen_hammond to col-jack-o'neill** is removed as it's a directional link in the wrong direction. The shortest link is "2".*

Use link weight (Remove links with negative values)

312

Use link weights finds the shortest path from A to B using minimal weights as opposed to minimal number of links. Also the shortest path is the total of the link weights — not the number of links total.
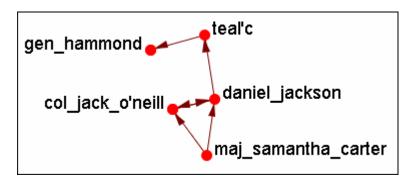
*The shortest path is from **col_jack_o'neill to teal'c (5)** then from **teal'c to gen_hammond (1)** for a total of "6".*

Use link weight (Convert negative links to positive values)
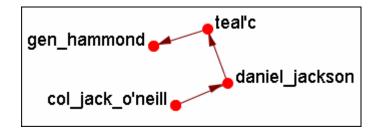
Shortest path: 3

Uses the **absolute** values of numbers. A **negative number** becomes a **positive number**.

Use link direction and link weight with absolute values

Shortest path: 3

Sphere of Influence/Ego Network

The **Sphere of Influence** tools helps show how nodes are interconnected by displaying links to other nodes. You can also control the size of the Sphere of Influence.
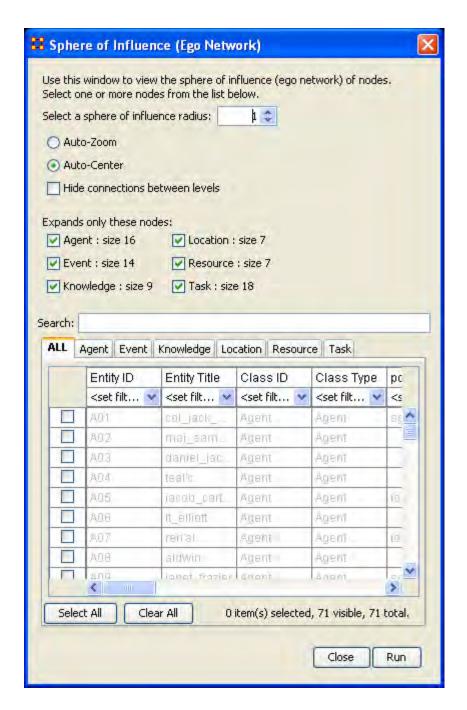
Sphere of Influence

Each node within a network has a unique **Sphere of Influence** or **Ego Network**. Essentially it's a direct relationship with it's neighbors as a function of specified path length. The ORA Visualizer allows you to focus on this relationship by creating an **Ego Map** centered on any particular node you choose. This mapping ignores directionalty and counts equally both in and out degrees it finds to the sphere surrounding the node. Cycles afre an andemic part of a Sphere of Influence and are not highlighted.
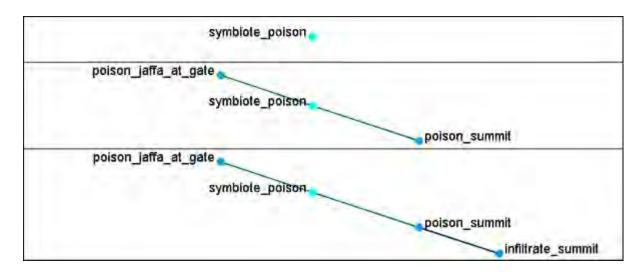
From the Visualizer's Main Menu select **Tools > Sphere of Influence**.

The following dialog box will then appear. From here you can select the **Size of your sphere's influence, the NodeSets, and Key Sets to use**.

Select a sphere of influence radius
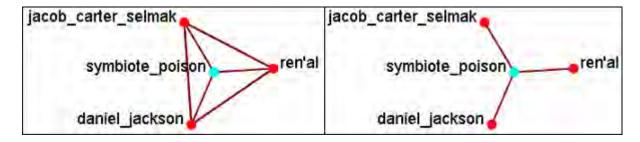
The drop down box allows you to choose the **path length** for a particular node(s). The value starts out at 0. Below is an example of a radius of 0, 1, 2, and 3. In many other cases the links will branch out as you increase the radius.

Auto-Zoom : Maximizes your ego network within the Visualizer.

Auto-Center : Keeps the visualized Ego Network centered within the visualizer display.

Hide connections between levels : Removes any connections between nodes that do not have a connection to any node in the filter.

Expand only these nodes : Allows you to select the components you wish to have displayed in a node's Sphere of Influence.
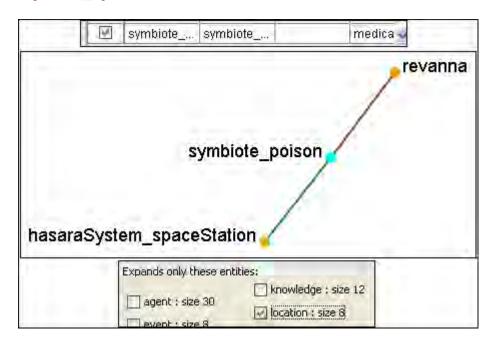
Filter Commands : Used for selecting the filters to use on the current visualization.

Node selection table : Allows the selection of individual nodes for visualization. First selected is **symbiote_poison** in the bottom table. Then the NodeSets **events and locations** were added. Only the nodes that are linked to the original **symbiote_poison** are displayed in the visualizer.
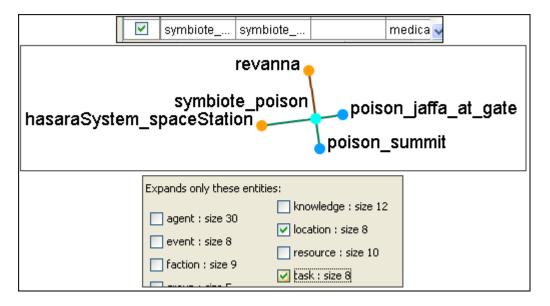
A Sphere of Influence Example

This example will show how to take one node and expand upon it to show how it is related to nodes in other NodeSets. First, from the table at the bottom, select a node. Here we select the **symbiote_poison**. From the **Expands only these nodes:** place a checkmark in **location**. It reveals

316
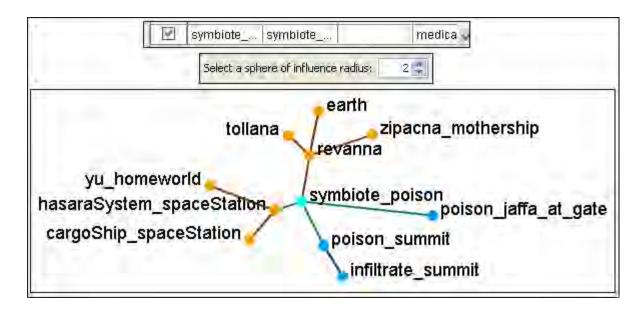
that symbiote_poison is connected to two locations, **revanna and hasaraSystem_spaceStation**.
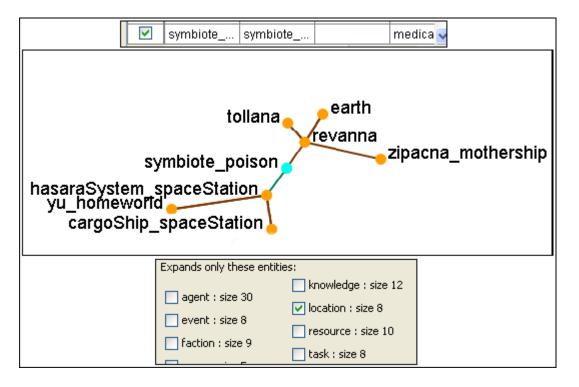
Next we'll place a checkmark next to **tasks:**. This reveals that there are two tasks that relate directly to symbiote_poison, **poison_jaffa_at_gate** and **poison_summit**.

Now that we've got a couple of nodes on the screen let's increase the radius of the sphere to 2. The tree will now display all nodes within 2 links of symbiote_poison.

Lastly you can eliminate any NodeSets by removing them from the list. Remove the checkmark from **tasks:** to reveal only the locations up to two links away.
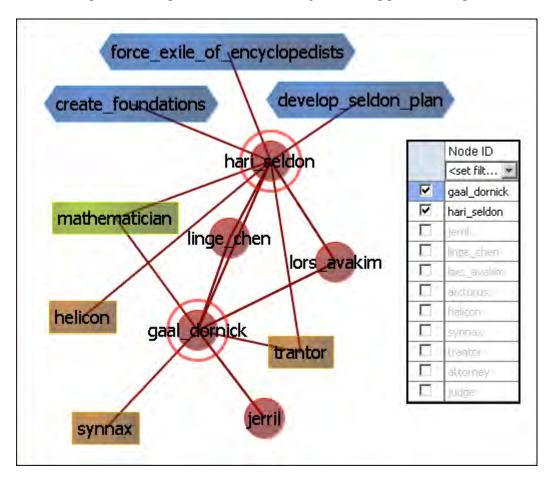
The **Sphere of Influence** allows you to hone your search and show
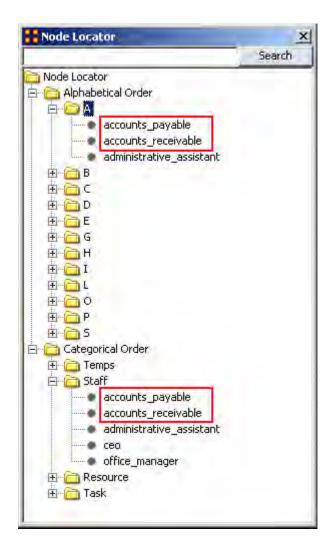
 Rings

What are node rings in the visualizer?

Rings help you keep track of which nodes have been selected in a Sphere of Influence. Right-Clicking on a node allows you to toggle the ring on and off.

They're not much more than rings around a node but rather they are used to help accentuate node(s) amongst other nodes. Currently, the only use they have is for Sphere of Influence, where the node(s) whose Sphere of Influence we're looking at get a ring around them.

Node Locator

The **Node Locator** is a tool palette that allows you quickly zero in and find a node that may be obscured in a complex visualization. Essentially, when you click on a node in the Node Locator tool, the Visualizer will then pan over to the node centering on it. Below are procedures for using the node locater.

From the ORA Visualizer: **Tools > Node Locator**. This brings up the Node Locator dialog box. It contains to folders in it's root, **Alphabetical Order** and **Categorical Order**.</td

Alphabetical Order : Contains one folder for each set of nodes divided into letter groups. All nodes beginning with the same letter will contained in one folder.

Categorical Order : Divides the nodes into node types. There were four NodeCLasses in this example and it created four folders under Categorical Order.

Notice how the nodes for Accounts_Payable and also for Accounts_Receivable appear in both areas.

Selecting a Node

Whenever you select a node in the Node Locator dialog box that node will be highlighted. Since it's hard to see with a white background this example uses the black background. Notice the white circle around the accounts_payable node denoting it has been selected. If you select a new

node the previously selected node will return to normal and the new node will be highlighted.

Micro Simulation

Micro Simulation allows you to run some basic simulations on the curretly loaded visualization. LInk weights are important as once they are normalized, they correspond to the probablity of a transmission.

General Thoughts on MicroSims

The simulations themselves are not that complex. Typically Disease, Money, and Information can be described as **types of flows**. At any given timepoint, any entity that has one of those things are going to try to give it to another node, with each flow having their own rules.

Disease : An entity can not re-get the disease, and an entity can only give out the disease while it has the disease. The entity can also give out a disease to multiple other entities. After a user-specified number of time periods, the entity loses the disease.

Money : Once an entity gives away the money, he cannot give the money to anyone else. There is no limit on re-getting the money.

Information : Once an entity gets information, they cannot lose it. They also never stop giving out the information.

Technology Adoption : How quickly a new technology might spread through a network.

It's important to understand how these transmissions occur. During the **transmission** phase, it looks at every node that has a resource. Then it iterates through all outgoing links trying to give that resource through that link (in the case of money, since it can only go through one link, we randomize the order links are tested). When looking at a link, it first does a check to see if the transmission can occur based on the link weight. Then it does another CHECK to see if the transmission can beat the transmission resistance.

321

These transmissions continue based on the rules outlined above for the user specified number of time periods. Then it generate a dynamic meta-network that shows for any given time period what entity was associated with a resource.

Diffusion of Ideas
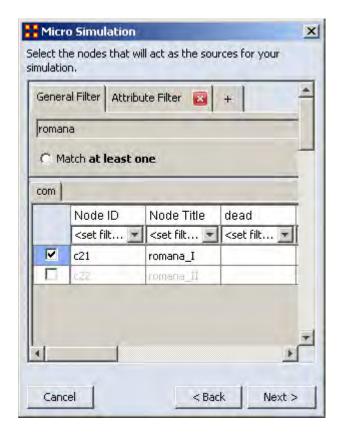
In the drop down box you are prompted to selected a **Square Network** for the simulation. If the NodeClass you select is in multiple square networks, then all will be displayed below. Place a check mark next to the network to use.

This will create a new meta-network consisting of a single node class with the same nodes as the input node class, and a single network with the same links present in the selected networks. In the event multiple networks are selected and multiple links from node A -> node B are present, the resulting link from A -> B has a weight equal to the sum of all A -> B links.

Click [**Next**]
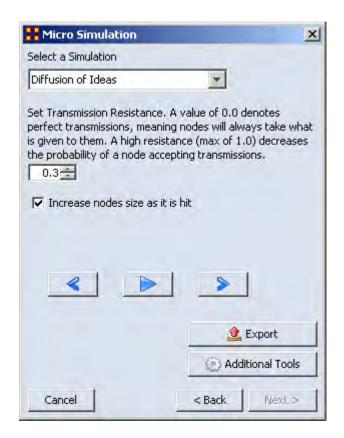
In the Node Selector box you chose a node(s) where an idea begins. I typed **romana** into the search box which displayed two choices. I placed a check mark to chose the **romana_I** node. When you've made your select, click [**Next**]

Two main controls

Transmission Resistance : ranges from 0.0 to 1.0. The higher the resisitance the less likely a node will accept what is offered. Setting this to [**0**] allows for perfect transmission and the target node will accept whatever is offered.

Increase nodes size as it is hit : helps in seeing which nodes have been activated. Nodes get progressively bigger with each hit.

We'll leave the Simulation to **Diffusion of Ideas**. Say we don't want ideas to spread automatically but want the nodes to have a little resistance. So the Transmission Resistance it increased to 0.3. I also placed a check mark in the **Increase node size as it is hit** box so it will show the amount of hits a node gets.

When you are finished, click [**Next**].

Dispersion of Money

This mode asks for the amount of **money units** to start with then makes them travel around the network. The amount of money stays constant.

An agent can give away money it posses (all or nothing).

Once given, an agent cannot give away the same money.

An agent can re-acquire money previously given away.

There is no limit to the amount of money an agent can receive/hold.

Agents pick randomly the order of neighbor(s) to give away money to in effort to reduce endless loops.

Information

An agent can give away information it has access to.

An agent retains information even after giving it away.

An agent never looses information it gains.

An agent never stops giving away information.

Adoption of Technology

The Transmission Resistance determines how easily technology is adopted by the network. This is a unique simulation. A node is more likely to adopt a piece of tech if it has many neighbors trying to convince it to adopt the tech. A node is more likely to un-adopt a piece of tech if the user has fewer neighbors also using the tech.

An agent may adopt technology if enough of its neighbors use the technology.

An isolated agent will flip a coin to adopt a technology.

An agent can stop using a technology, especially if its neighbors are not using the technology.

An agent can re-start using a technology.

An agent has a user-adjustable level of resistance to using **new** technology.

Diffusion of Disease

This function has two additional parameters:

Number of periods nodes remain contagious : After a node gets infected it will stay infected for a set number of periods before it then becomes immune. After that it can no longer spread the disease.

Percent nodes immune at start : You can set a percentage of nodes to be immune from the start. These nodes will be colored yellow.

Other information for disease simulation.

An agent can not re-get the disease.

An agent can only give the disease while it is infectious.

An agent can give the disease to multiple other entities

An agent loses the disease after a user-specified number of time periods.
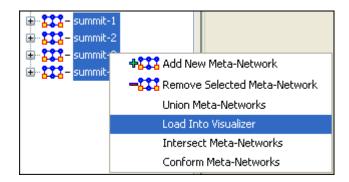
View Networks Over Time

The **Networks Over Time** loads multiple Meta-Networks into the Visualizer and displays them as **Time Slices**.

NOTE : Data that is not dated will be loaded in the order it was selected, not the order it is in the tree and not the order it was loaded in. This only happens when the data has no dates associated with it.
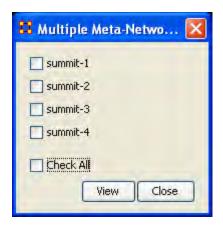
Procedure

Highlight the Meta-Networks to load into the Visualizer. For this example I divided the whole Stargate Summit network into four distinct phases. 1) The meetings; 2) What was supposed to happen; 3) What did happen; 4) The attack on the Tok'ra homeworld. These are Summit 1-4.

Start the Visualizer. Then select **Tools > Networks Over Time**

The dialog box appears to verify you've chosen the correct Meta-Networks. Place a checkbox in the ones you want or **Check All**. Then select **[View]**.

To control the Over Time function from the main menu, select **Tools > Networks Over Time**.
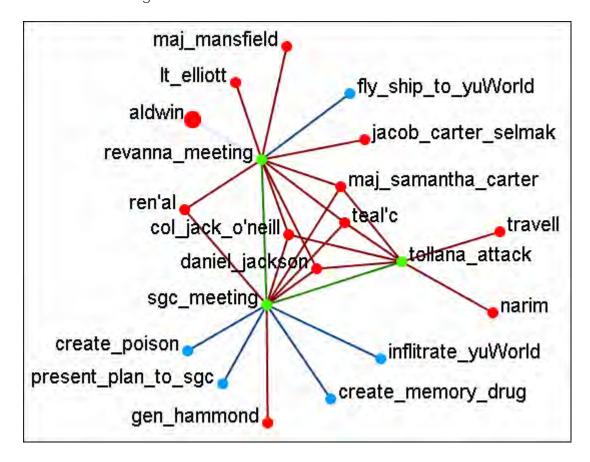
The **Phase Duration** has two controls. **Transition Phase duration** sets the amount of time in milliseconds for the fading out of the previous set and the fading in of the new set. The **Display Phase duration** sets the amount of time in milliseconds that a set will be displayed before moving to the next set.

Constant maintains a node's position throughout all Timeslices. **Determined by Timeslice** allows you to move a node independently in each Timeslice.

Record Node Locations upon Exiting a Meta-Network maintains a node's position when changing Timeslices. **NOTE:** It does not maintain the node's positions after you exit the Visualizer. Upon returning to the Visualizer all nodes will be in default positions.

The **Slider bar** controls which Timeslice is displayed. The buttons at the bottom, **[Play] and [Stop]** control the motion display.
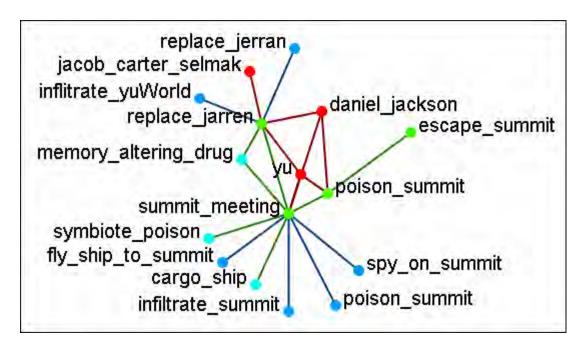
Summit-1

The initial meetings between the SGC and the Tok'ra

Note: You can select multiple nodes by simply clicking on them. You do not need to depress any other keys (i.e., ctrl+shift, etc.).
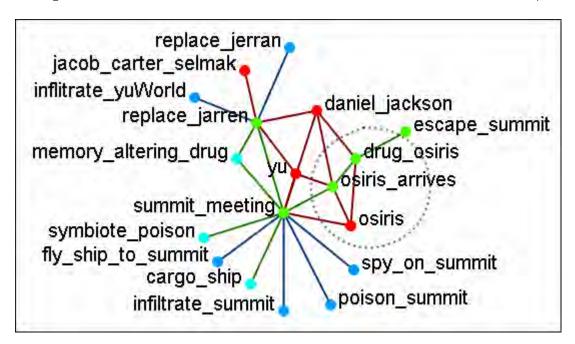
Summit-2

The plan as it was **supposed** to go
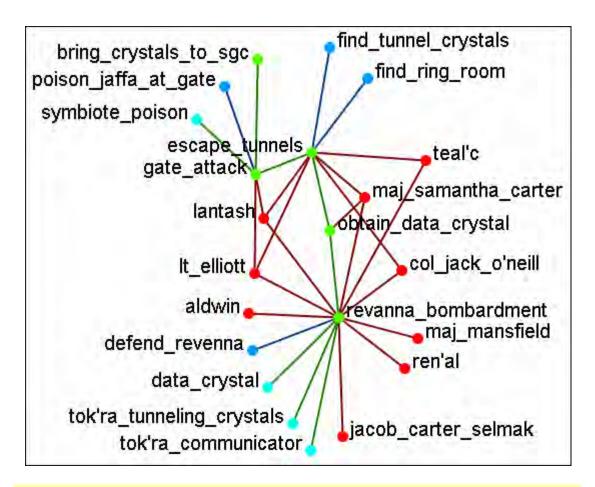
Summit-3

What really happened when Osiris showed up unexpectedly. Instead of poisoning the Summit daniel_jackson decides to forgo that action after hearing the information from Osiris. He's almost found out but still escapes.

Summit-3
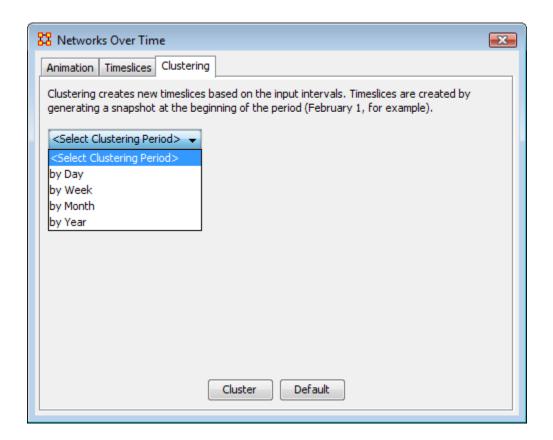
What was happening back on Revanna while daniel_jackson was at the Summit.

Time Clustering

Time clustering is available via the Over Time dialog in the visualizer. It's a fairly simple panel - there's a ComboBox to select which period you want to cluster by (by Day, by Week, by Month, by Year), and two radio buttons to select what a week begins with (only used when clustering by week). Finally, there's a simple button to cluster.

Clusters are fairly straightforward. It looks at the start date (earliest date available in a set of timeslices) and the end, and grabs snapshots at the given interval, looking at the beginning of that interval. So, if you cluster by month, it'll move through each month, grabbing a snapshot of the 1st day. If no Timeslices exist for that day, it just uses the Timeslice with a date closest to that day without going over.

Visualizer Layout Menu

Run Active Layout : Turns on the motion and allows ORA to attempt to sort out the positioning of the nodes.

Stop Active Layout : Turns off the motion and stops all nodes from moving. This allows you to move individual nodes around without disturbing the rest of the Meta-Network.

Spring Embedded Layout : This layout operates fine in datasets of approximately under 500 nodes. Much higher than that and the layout gets sluggish. A single layout iteration could take two minutes or longer. Not really practical.

Spring Embedded (with enhancements for large data) : The enhanced layout with large datasets. It uses the same basic algorithm but has some optimizations built in. With 10k nodes or so, a single layout iteration takes 10 seconds or so.

NOTE : Now you're probably asking **"if this new algorithm is so good, why not keep it on all the time?"** Reason: If you pick up and drag a node, and as such drag an entire meta-network, the display will look jittery and confusing. So, whenever you load in under 500-nodes, the normal version is enabled. More than that, it uses the enhanced version.

Run MDS Layout :

Run Tree Layout :

Run Circle Layout (Pendants to Outside): Takes all pendants and puts them in a circle. All Nodes connected to a pendant are on a circle inside the pendant circle. Repeat until all nodes are in some circle, with the inner most nodes being the farthest from any pendant.

Run Circle Layout (Center is High Betweenness) : Calculate the Centrality-Betweenness for the graph. Take the node(s) with the highest value. Place them in the center. All nodes connected to those nodes get put on a circle surrounding the center. Repeat as necessary.

Run Hierarchical Layout : Tries to layout the network in a pyramid formation so that all arrows either point up or point down.
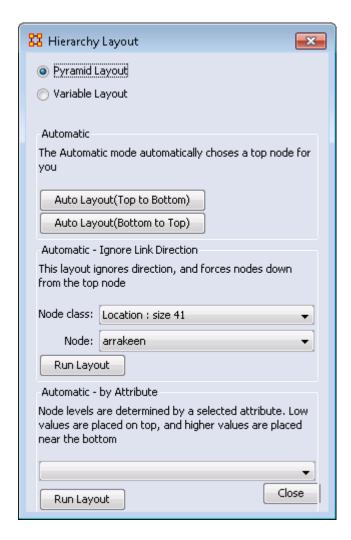
Run Box Layout : More of a Polygon Layout, really. You select 1-4 groups of nodes (either by node type or by node class). Each group is then laid out as one edge of a polygon. 1 Group = circle, 2 Groups = 2 parallel lines. 3 Groups = Triangle, 4 Groups = square

Attribute/Measure Layout : Uses the input attribute and/or measure as the x,y coordinates of the nodes.

Hierarchical Layout

Run Hierarchical Layout :
The Hierarchical layout option allows for two different layout styles.

The **Pyramid style** looks at level 0 as being the peak of the pyramid. Level 1 being the one beneath that, and so on. A level's width is equal to the level number (with the distance between levels being a constant). This gives it the appearance of a pyramid. This is the one that's been implemented for ages now.
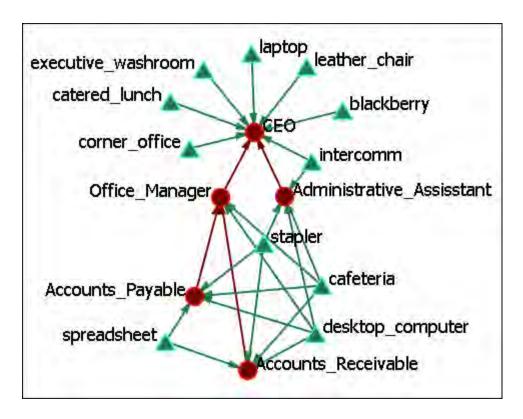
Variable style Variable Style is similar in that the distance between levels is still constant, but the width of a level is now variable. Instead of being based on the level number its based on the number of nodes in that level. If you look at top-heavy networks, you'll learn the appeal of this layout.

You will be prompted for a square network (including the Visible meta-network, since technically it's square), and then for a node to use as the **peak** node.

Finally, you must choose the desired build direction. This refers to Link direction, since a link could mean **reports to** just as easily as it could mean **commands**.

NOTE : This layout typically works well with **non-circular** networks. i.e. networks were links have links going only one direction. Using circular layouts can create problematic displays.

Regular Layout : showing links between resources and the office staff.
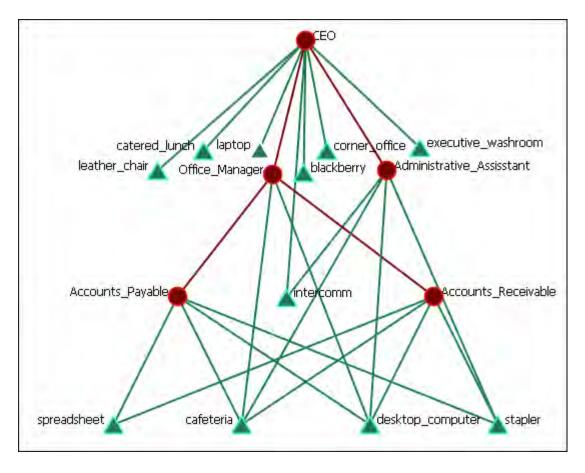
Automatic

The **Automatic** buttons will take the most direct path in creating a hierarchy. Whichever node has no links out will most likely become the top node. You can chose [**Auto Layout (Top to Bottom)**] or [**Auto Layout (Bottom to Top)**].
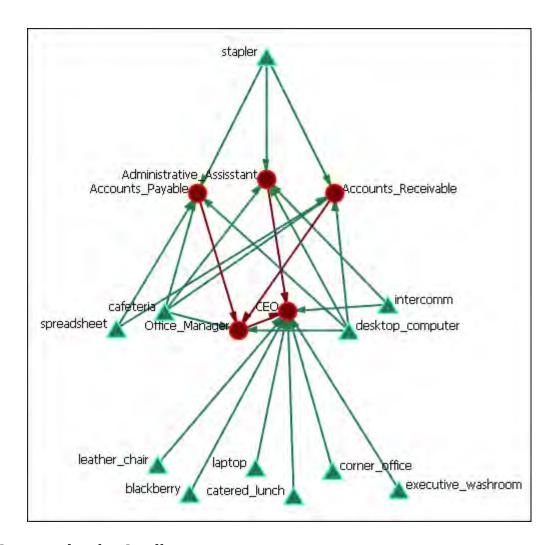
Arrows always try to maintain their direction meaning, in the event of cycles, for example, we try to minimize the number of arrows that go "against the current", so to speak. It will also warn you should you try to input a symmetrical network, as those don't really have **bosses** or **subordinates**.

Hierarchical Layout (Top to Bottom) : which places both staff and resources in a top to bottom pyramid. This view makes it much easier to detect which resources are used by which staff member.

Automatic - Ignore Link Direction

Automatic - Ignore Link Direction Layout : Allows you to chose the node you want at the top. Notice in the image the direction of the arrows are not in the same direction.
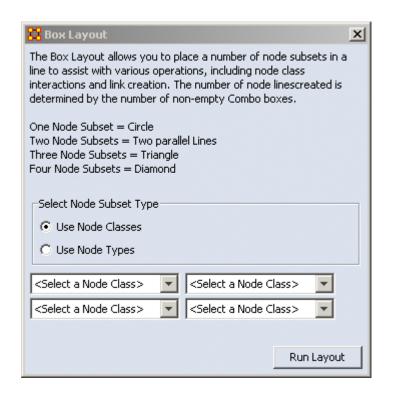
Automatic – by Attribute

This layout is based on the attributes of the ndoes. It requires all relevant nodes to have **some number-based attribute**. The lower the value of the attribute, the higher in the hierarchy it will be. Think of 0 as being the top…level 1 is one below, and so on.

 Box Layout
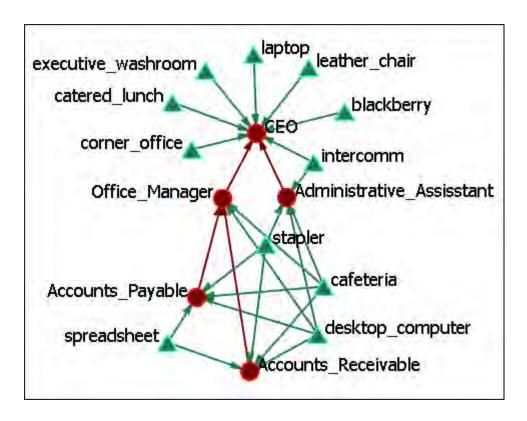
Run Box Layout : This layout allows you to chose from one to four Node SubSets and will arrange they depending on how many are choosen. The default is to use individual Node Classes. i.e. This would allow you to choose different agent Node Classes for different sides of the shape.

Use Node Types

Using this option allows you to choose all Node Classesat the same time, i.e. if you have multiple Agent Node Classes then choosing Agent would select all of them for a single side of the shape.
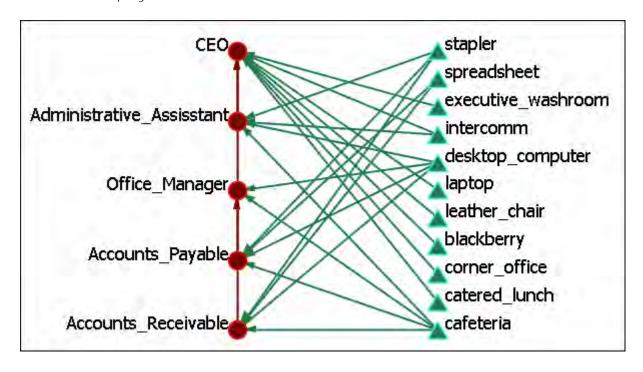
Regular Layout

Here's a Meta-Network as originally display in ORA.
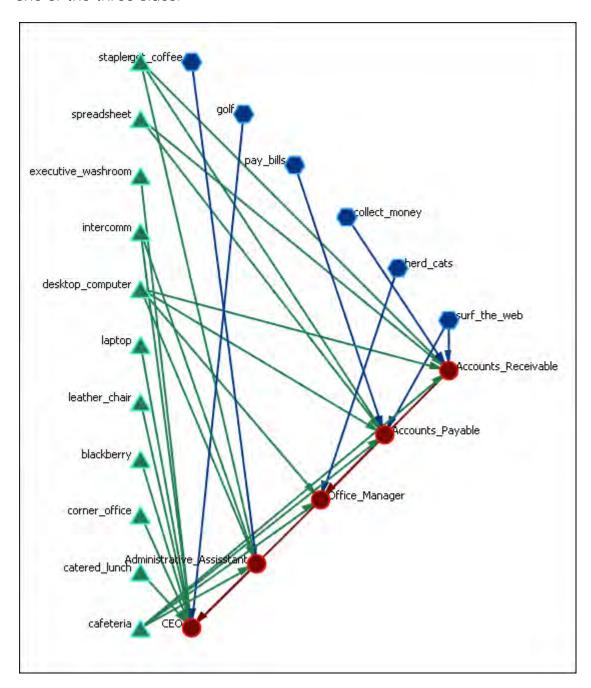
Box Layout (Two NodeSets)

Running the Box Layout lines up the Node Classes opposite each other with their links displayed between them.

Box Layout (Three NodeSets)

Adding a third Node Class would form a triangle with each Node Class on one of the three sides.
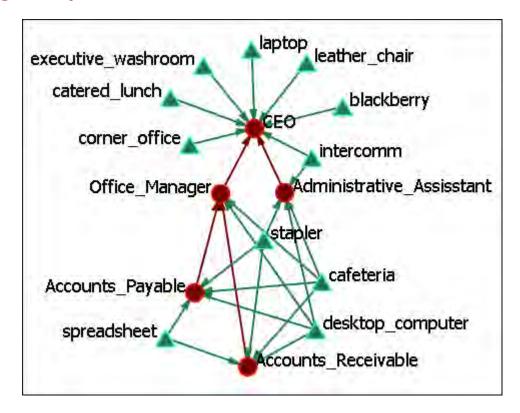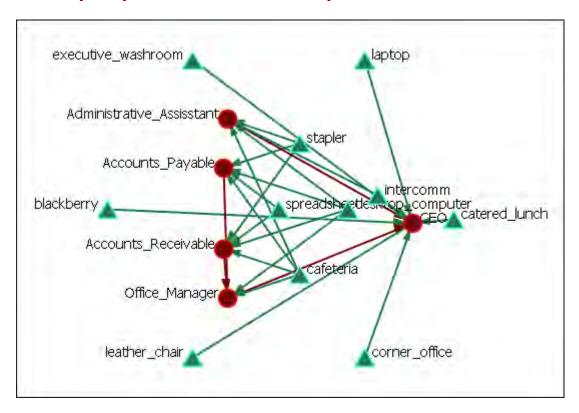
Circle Layout (Pendant to Outside)

Run Circle Layout (Pendants to Outside) :

Regular Layout

Circle Layout (Pendants to the Outside)
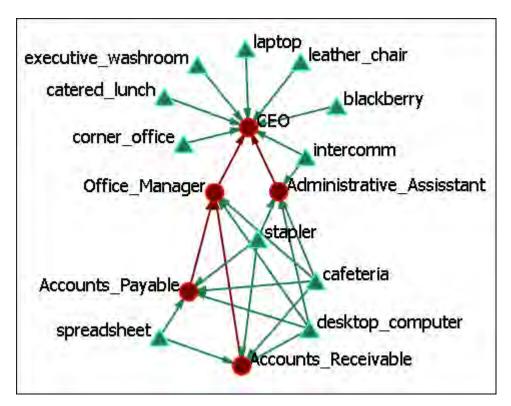
Here we see the spreadsheet and the desktop computer are at the center of the circle as they have multiple connections. You can also see the pendants (executive washroom, laptop, catered lunch, corner office, leather chair, and blackberry comprise the outer circle. Lesser connected nodes drift to the outer circle.

Circle Layout (Center is High Betweenness)
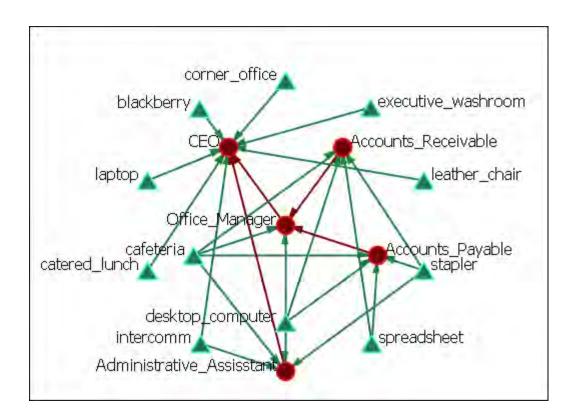
Run Circle Layout (Center is High Betweenness) :

Regular Layout

Circle Layout (Center is High Betweenness)

This example shows how the Office Manager is the most connected node being responsible for two nodes under him and reporting to the CEO.

Display

Link Appearance

Link Labels > Show Link Labels : Toggles on/off the display of the labels of the links.
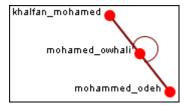
Link Labels > Show Weight Value Labels : Toggles on/off the display of the weight of the links

Link Labels > Show Weight Sign Labels : Sets the label and shows whether their weight is positive or negative.

Link Color > Default : Removes all custom link coloring and resets all links to default.

Link Color > By Link Weight : You can set a range of colors to represent the weight of links. A dialog box allows you to chose the colors for the minimum and maximum weights.

Link Color > Match Source Node : Colors the link the same as the source.

Link Color > Match Target Node : Colors the link to match the target.

Retain Bidirectional Links : When two nodes have links to each other only one link will be displayed.

Split Bidirectional Links : When two nodes have links to each other only both links will be displayed. These links will be rendered as curved links between the nodes.

Show Links : Toggles on/off the display of links between nodes.

Scale Link Width to Weight : Toggles on/off the display of links between nodes.

Scale Link Width to Number of Links Between Nodes : Adjusts the width of the link to represent the weight of the link.

Show Arrows : Toggles on/off the display of the arrows of the links.

Show Self Links (Loops) : Shows whether there is a connection between a node and itself. It is designated with a circle pointing back to itself.

Node Appearance

Label Style > Default : Sets the label to appear to the right of the node. This is the default position for labels.

Label Style > On Right of Node : Sets the label to appear to the right of the node.

Label Style > On Left of Node : Sets the label to appear to the left of the node.

Label Style > On Top of Node : Sets the label to appear on top of the node.

Label Style > On Bottom of Node : Sets the label to appear beneath the node.

Label Style > Centered on Node : Sets the label to appear directly centered on the node.

Node Style > Toggle Node Size : Rotates between the three sizes and styles for the display of the nodes.

Node Style > Spot : Sets the size of all nodes to a small colored circle.

Node Style > Small Shapes : Sets all nodes to a colored shape (circle, square, triangle, pentagon, diamond, hexagon).

Node Style > Large Shapes : Sets shape of nodes to a colored shape only large enough to contain the label.

Node Color > Save Node Color To Attributes : Creates an attributes to store the current color of all nodes. If the color for a nodeset is changed by coloring by attribute, coloring by measure, or changing the color in the Legend dialog then this function will save that information.

NOTES : Meta-Network must be saved in the Editor in order for colors to be reloaded the next time the Meta-Network is loaded.

The NodeClasses retain their default color in the Legend dialog. i.e. all agent NodeClasses will default to red.

344

Node Color > Load Node Color From Attributes : If a Meta-Network contains color information for the nodesets then this function will load the color into the Meta-Network and reset the display.

Node Color > Show Node Occurrance Transparencies :

Node Color > Color Nodes by Attribute or Measures : ALlows you to chose an attribute or measure to run on a Meta-Network and will color the nodes in specific groups.

Node Color > Color Nodes by Componant :

Node Color > Color Nodes by Concor Grouping : Divides and colors nodes according to the Concur Grouping

Node Color > Color Nodes by Newman Grouping : Divides and colors nodes according to the Newman Grouping

---

Reset Node Color and Size : Sets the color and size of all node to the default values.

Size Nodes by Attribute or Measure : Allows you to chose an attribute or measure to run on a Meta-Network and will size all nodes accordingly.

Apply Log Scale to Current Sizing :

Show Labels : Toggles the names of the nodes on and off.

Load Node Images : If nodes have image files attached to an attribute this will display the picture instead of the node icon.

Remove Node Images : Removes all images from the visualizer and returns all nodes to their icon status.

Node Shaper : Allows you to chose the size and/or shape of any set of nodes. The set is not just one particualr NodeClass but can be created from individually choosen nodes from any NodeClass.

Attribute Measure related Options :

Comments

Show Comments : Toggles on/off the display of comments

Set Comments Font : Sets the font used in Visualizer comments.

Set Comments Color : Sets the color used in Visualizer comments.

Title

Title Location > Top : Sets the Title to the top of the Visualizer window.

Title Location > Bottom : Sets the Title to the Bottom of the Visualizer window.

Title Location > Left : Sets the Title to the left of the Visualizer window.

Title Location > Right : Sets the Title to the right of the Visualizer window.

Title Location > Center : Sets the Title to the center of the Visualizer window.

Set Title : Create a title for a screen shot. The title will remain a static size and retain its position regardless of the zoom of movement of the display.

Set Title Font : Set the font used in the title of the screen shot

Show Title : Toggle visibility of the title.

Allow Click to Edit Node Title :

Show Mouse Overs : Toggles the information boxes on and off whenever you hover the mouse over a node or link in the visualizer.
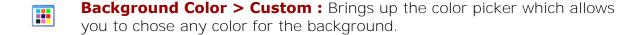
Autozoom : Zooms the display either In or Out to bring all nodes into view.

Background Color

Background Color > Black : Sets the Visualizer background to black.

Background Color > White : Sets the Visualizer background to white.

Background Color > Custom : Brings up the color picker which allows you to chose any color for the background.

NOTE : Some of the possible background colors can make it difficult to view the visualizer. Your best choice is black (although white is used in the examples for easier printing).

Background Image

Background Image > Load : Allows you to chose a **.jpg, .gif, or .png** to use as a background.

Background Image > Clear Image : Removes the image and returns the background to the last background color used.

NOTE : Some pictures may make viewing the network somewhat difficult.

Black and White Only : Removes all colors form the visualizer.

Grayscale : Removes the hue from all nodes and links displaying them as shades of gray. If any of the nodes are difficult to read try changing the background color to a darker or lighter color. A black background works very well for grayscale images.

Reset Node/Link Options : Resets all node and link colors back to default.

Link Color
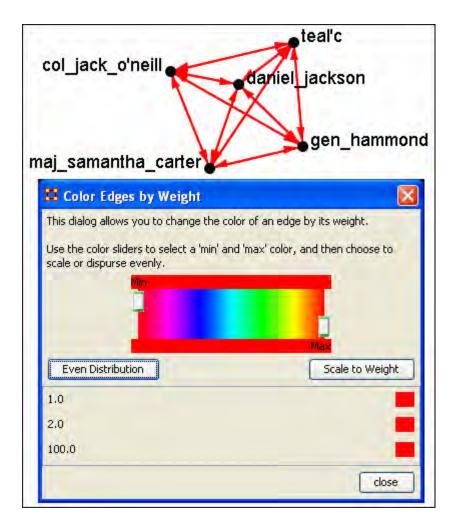
In ORA you can color links according to several preferences. Below are the procedures for doing so.

	col_jack_o'neill	maj_samant...	daniel_jackson	teal'c	gen_hammond
col_jack_o'neill	0.0	1.0	2.0	1.0	2.0
maj_samant...	2.0	0.0	1.0	2.0	1.0
daniel_jackson	1.0	2.0	0.0	1.0	2.0
teal'c	2.0	1.0	2.0	0.0	1.0
gen_hammond	100.0	100.0	100.0	100.0	0.0
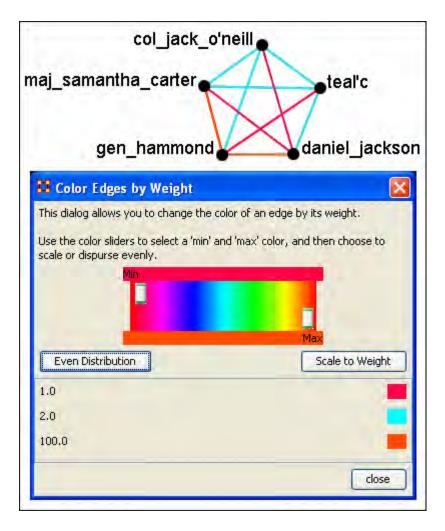
NOTE : For this example all the nodes are colored black so as to decrease the color confusion.

NOTE : For this example the link weights have been drastically altered from the original values to assist in display purposes.

NOTE : Below is a warning for the placement of the **Min** and **Max** sliders. When the sliders are set to the absolute left and right positions this causes all the colors to be set to red. Move the sliders in to prevent this.
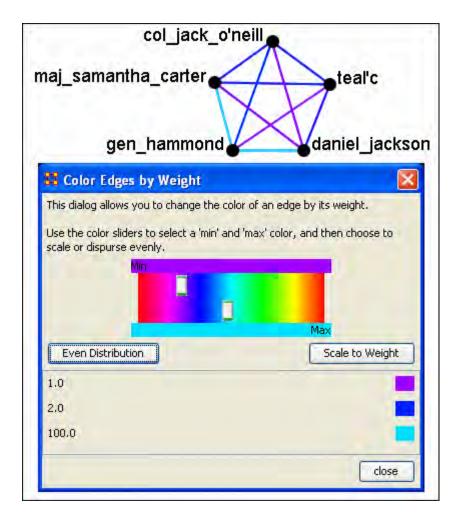
Even Distribution - The Full spectrum
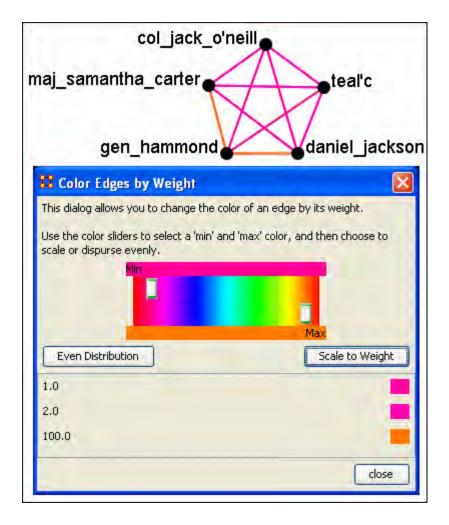
The colors used are evenly distributed across the chosen spectrum. Using a NodeSet with weights of 2, 4, 6, 8, & 10 would have three colors chosen at evenly spaced intervals across the spectrum.

Even Distribution – The Limited spectrum
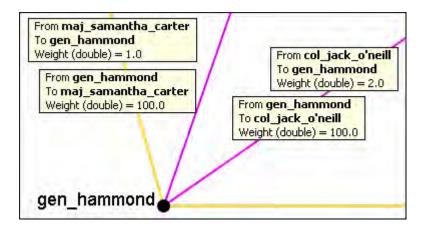
Limiting the spectrum to smaller set will cause some of the numbers to contain very similar colors. This might make it difficult to tell the various shades of blue apart. Use this only when there are a very limited number of values.

Scale to Weight

When scaling to weight the **100** values cause a spread to occur that makes all other values in the same general color area.

Link Values

When a link is bi-directional and the weights of the links are not the same the Visualizer will color the link that was drawn last.

Change link Color

After setting the type **Even Distribution/Scale by Weight** the color of the links can be set manually through the color changer. Click on the color box of any of the links to bring up the color change dialog box. Select a new color and select **[OK]**.
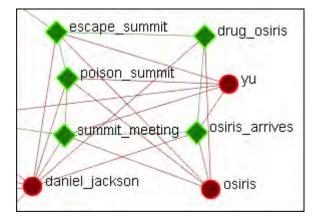
Comments

While in the Visualizer **Comments** can be added to the display. Right-click a node and from the contextual menu select **Add Comment**. The comment is placed in a colored box with a dashed arrow connecting them to the original node.

NOTE : These comments are not saved when exiting the Visualizer.
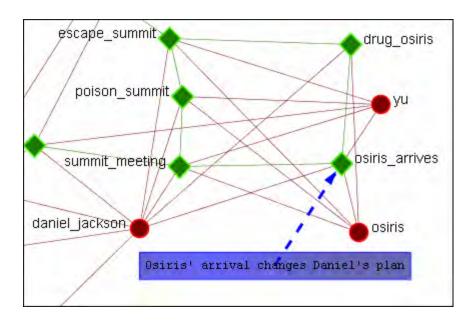
NOTE : The main use of these comments would be in creating screen shots for display in order to add commentary for a report.

Below is a section of the Stargate scenario with the unexpected appearance of Osiris.
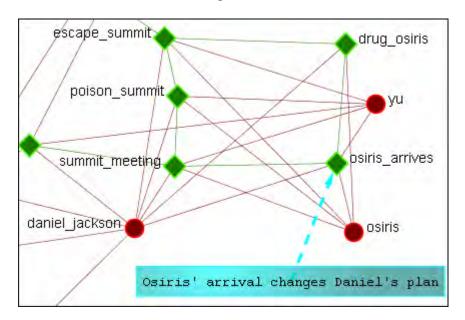
Adding a Comment

A comment is added stating Osiris's arrival changes the plan.
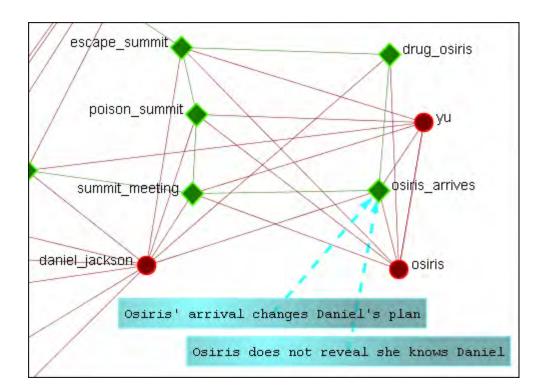
From the main menu, under the Tools item, the font size and color, as well as the color of the box, can be changed.

Adding Multiple Comments

It is also possible to add multiple comments to the same node.

Deleting a Comment

A comment can be deleted by first selecting that comment. A right-click brings up the contextual menu. From this menu select **Delete Selected Node(s)**.

 Mouseovers

NOTE : To access the contextual menu the option must be checked.

Go to **Display >** **Show Mouse Overs** and make sure the option is check marked.

When the pointer is passed over any Node in the visualizer a **yellow tooltip box** will appear showing information on links to other nodes as well as values for measures than have been run. Below is a screen shot comparing two mouseovers.

355

If an **Attribute** is eompty ORA will leave a blank line for that attribute. Only if the attribute has a value will it be displayed. **Neighbors** are all the links to other nodes.

Mouseover Display Limit

With the mouseover display active ORA will attempt to display which node or link you have the pointer hovering over. But there is a limit of **5,000** links when ORA no longer attempts to dusplay the link. It will, however, still attempt to dusplay which node the pointer is hovering over.

Node Images

Nodes can be replaced with various symbols or pictures to make it easy to identify individual nodes. There are a few items to note.

All images should be reduced to approximately **50 x 50 pixels** before using in ORA. Bigger images will be reduced in size on the display and the image quality may possibly be degraded.

If a picture is placed on a background with the color **R=255, B=0, G-192** the background will be changed to the color of the node.
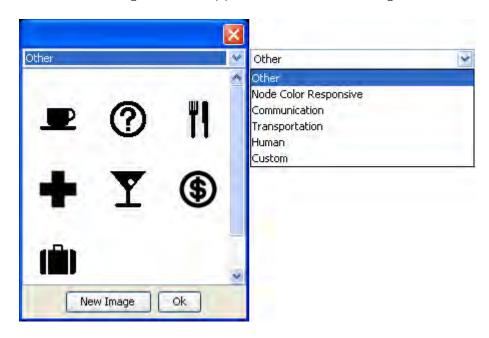
Setting a Image Folder

The Set Picture option is designed with modularity in mind, with adding new images as easy as putting a folder with images in the

include\OraFiles\images\visualizer\node folder. The dialog will automatically read that folder for subfolders, use the subfolder names as categories, and the images inside the subfolders as the images for each category.

Set Picture

Right-click any node and select from the contextual menu **Appearance Submenu > Set Picture**. The Node Selector dialog box will appear. The drop down menu contains various types of node images. With a node selected a custom image can be applied. Nodes with Images will scale.

Custom

The **Custom** option allows for the inclusion of a user's own icons.
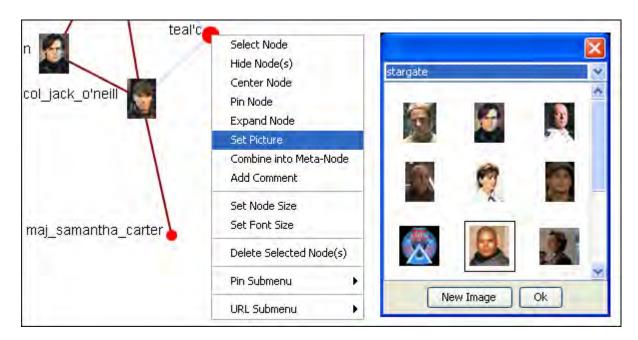
The custom category is used for images that have been previously loaded in the current meta-network. Clicking OK applies the currently selected node's image to all nodes selected in the visualizer.

Additional information can be found in the **Node Shaper** section.

The **[New Image]** button brings up a navigation box from which to navigate any directory and select an image file. The picture types can be **jpg/jpeg, gif, or png**. These will all be collected into the **Custom** section.

NOTE : Remember to reduce the picture size BEFORE importing it into ORA.
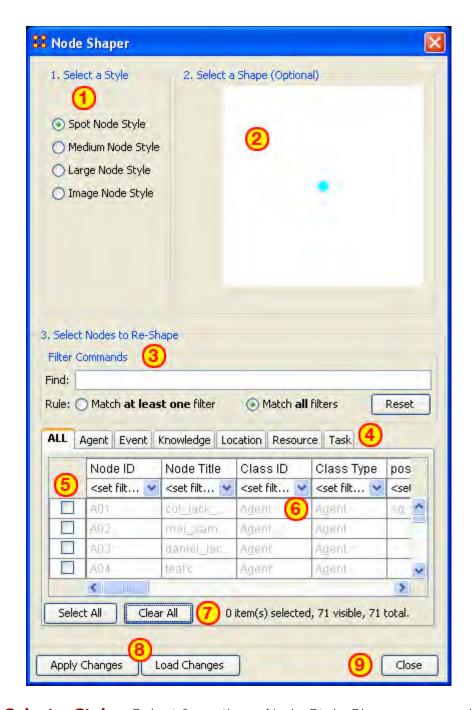
To set a new image right-click the node and select **Set Picture**. This brings up the picture dialog box. Locate the appropriate picture and select it. A black border will appear around the image. Select **[OK]** will set the picture to the node.

Node Shaper

The **Node Shaper** allows you to change the size of individual node(s) or assigning pictures. Below is an image of the Node Shaper widow and a description of how to change the shape of the nodes in your model.

1. **Select a Style :** Select from three Node Style Sizes or use an image in it place of the node.

2. **Select a Shape :** An optional shape can be used instead of the NodeSet's default shape. The node will retain it's default NodeSet color.

3. **Select Node to Re-Shape :** The bottom section contains the controls for selecting which nodes to work with. Typing anything in the **Find** textbox will automatically filter out non-matching nodes.

4. **NodeSet tabs :** Select which NodeSets to view. **ALL** shows every node or the tabs will show only their individual NodeSets.

5. **Check boxes :** Place a checkmark in the box next to the nodes you want to change.

6. **Filters :** Use for filtering based on individual attributes. You can filter on more than one attribute at at time. An example can be found in **Create a new Meta-Network**

7. **Select/Clear All buttons : Will either Select or Clear all VISIBLE nodes.** If a check marked node has been filtered out and is currently **NOT** visible when the **Clear All** button is pressed, then that node will retain it's checkmark and be affected by future changes.

8. **Apply/Load Changes :** After any changes have been made to the top portions selecting the **[Apply Changes]** buttons will cause them to be displayed int he Visualizer.

9. **Close :**

How The Node Shaper Changes The NodeSet
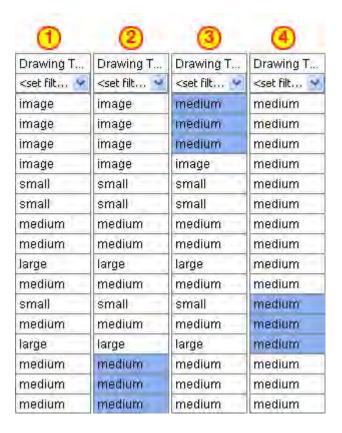
What follows are examples of how the **Node Shaper** changes the **NodeSet**.

Whenever the **Node Shaper** is called and a node is assigned something other than it's default shape new attributes will be added to **Every** NodeSet, not just the NodeSet you're working with.

- **Drawing Type :** This will define what type of node is to be drawn, shape of image. There are three options for size (small, medium, and large) and one option for image.

- **Nodes Shape :** There are four large shapes (circle, rectangle, ellipse, and hexagon), eight medium shapes (circle, hexagon, down triangle, pentagon, diamond, square, up triangle, and octagon), and one small shape.

- **Image :** Contains the name of the file associated with that node.

Changing one node will **create these three attributes** and place the default value in every node in every NodeSet. Any of these attributes can be manually changed from within the Main Interface Editor.

This is useful if you have changed many nodes and want to restore them all to their defaults quickly and easily. The quickest way is using the **Copy and Paste** method.

Column "1" contains the modified values. In column "2" three values of medium are highlighted and copied using **Ctrl-C**. In column "3" the cursor is moved to the top cell and **Ctrl-V** is used to paste those three values in. In column "4" this is repeated until all values are back to their defaults.

If the **Image** attribute contained a filename then changing **Drawing Type** to image will relink that file to the picture. Any of the four **Drawing Type** values can be placed in this column.
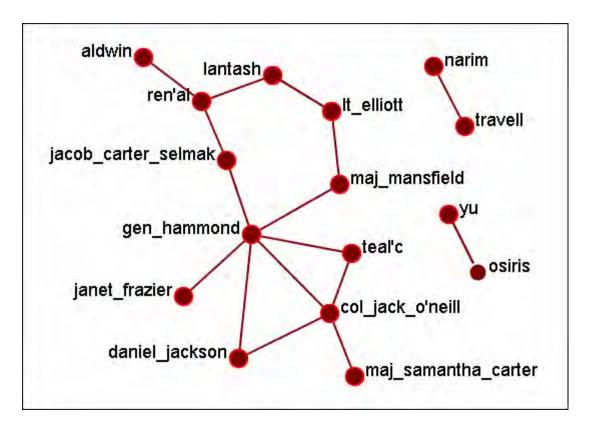
Node Shaper Examples

Original agent x agent Network
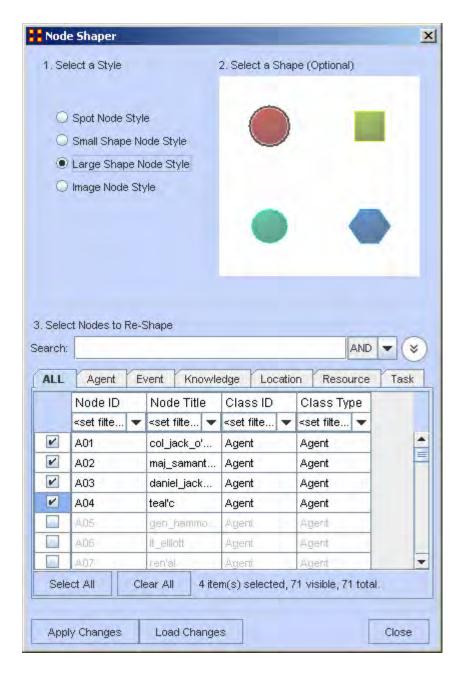
Below is the **agent x agent** network in the normal **medium sized nodes**.

We start out with the stargate agent x agent nodeclass set to medium size nodes.

361
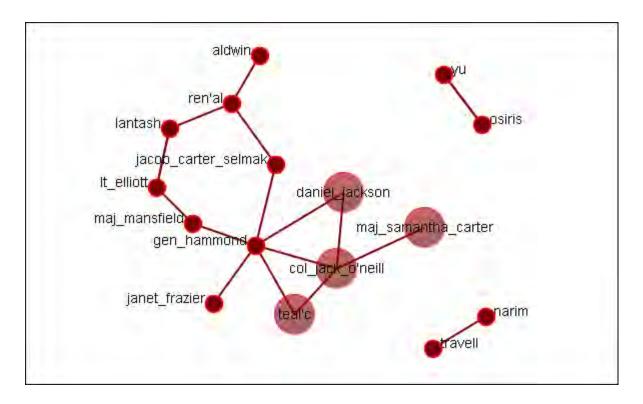
Changing a few nodes

From the Visualizer menu select **Display > Node Appearance > Node Shaper**. Click the tab marked [**agents**]. Place a check mark next to **col_jack_o_neill, maj_samanatha_carter, daniel_jackson, and teal_c**. Select the **Large Node Style**.
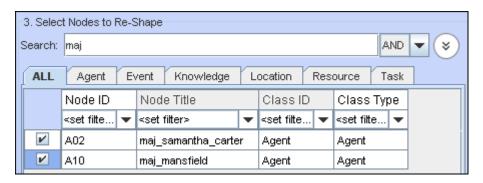
When you are finished click the [**Apply Changes**] button.

Now the four members of SG-1 stand out represented by larger nodes.

Select Nodes to Re-Shape

You can also use the Search box, in conjunction with the tabs, to bring up specific nodes in the grid. It's predictive so as you begin typing the grid will change. Continue typing until you see the node(s) you need. I wanted the two characters with the rank of major and knew their rank was a prefix. I typed **maj** into the search box and the two nodes were singled out.

![Visualizer Contextual Menus icon] **Visualizer Contextual Menus**

The Contextual Menu is displayed by right-clicking an a Node or Link. It contains many standard uses in manipulating nodes and links.

Node Menu

Pin SubMenu :

Pin All Nodes : Globally pins all nodes. Nodes are unaffected by the **[Play]** button or the movement of other nodes.

Unpin all Nodes : Frees all nodes from being pinned.

Pin Selected Node(s) : All selected nodes will become **pinned**

Unpin Selected Node(s) : All selected nodes will become **unpinned**.

URL SubMenu :

Attach URI : You can attach a URI to the selected node.

Go To URI : Activates your browser and goes to the URI associated with this node.

Appearance Submenu :

Set Picture : You can replace the node shape with a picture.

Set Node Size : You can numerically set the Node Size of an individual node.

Set Node Color :

Set Font Size : You can numerically set the Font Size for an individual node.

Enlarge : Increases the size of the selected node in the Visualizer

Shrink : Decreases the size of the selected node in the Visualizer

Brighten : Lightens the shade of the color of the selected node

Darken : Darkens the shade of the color of the selected node.

Hide Labels : Hides the label of the selected node.

Show Labels : Reveals the label of the selected node.

Show Ring(s) :

Hide Rings(s) :

Visibility Submenu :

Hide Node(s) : Removes the selected node (along with all links) from view in the Visualizer.

NOTE : This does not delete the node from the Meta-Network.

Hide Adjacent Node(s) : Hides all nodes that are linked to the selected node. The selected node remains visible.

Other :

Color Adjacent Edges : Allows you to chose the color for all links associated with the selected node. Only works on one node at a time.
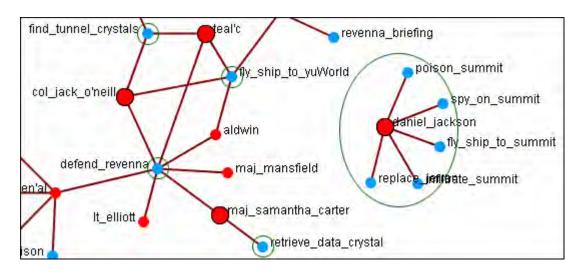
Select Node : Registers the node as **selected** when Ora affects multiple nodes.

Center Node : Brings the selected node to the center of the visualizer.
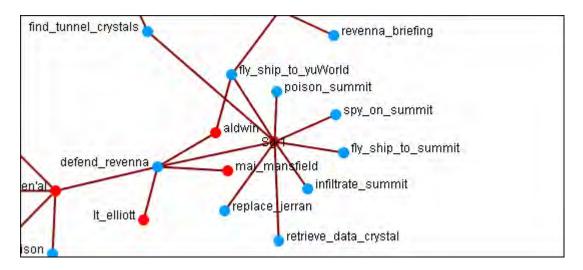
Pin Node : Pins node to background so it is unaffected by the movement of other nodes.

Expand Node : Shows all nodes and links connected to the selected node.

Combine into MetaNode : Multiple nodes can be selected and combined into a **Meta-Node** which will contain links to all the nodes that the individual nodes had. The four large circles of SG-1 are selected. The and green circles show which tasks are associated with these agents. Press, and hold, the Control key while selecting each node for the Meta-Node. Then **without releasing the Control key**, right-click one of the nodes to bring up the contextual menu. Select **Combine into Meta-Node**.

After combining the **task nodes** that were linked to the individual nodes are now linked to the **sg-1 Meta-Node**.

Add Comment : Attaches a comment to the **currently selected node**. Good for making notes for display purposes. Comments can be toggled to be shown or hidden form the Display menu. The color and font of the comments are also controlled from the Display menu.

Spread Out : Finds the center for the currently selected nodes. Then calculates a ray from that center through the nodes current location, and pushes the node along that ray by (currently) 40 units.

Delete Selected Node(s) : Allows for the deletion of nodes form a Meta-Network from within the Visualizer.

Link Menu

Link Status : Brings up the Link Status box with information on its weight, the network, source and target nodes, source and target nodeset.

Hide Link : Hides the link. This does not delete the link.

Set Value : Set a new value for a link.
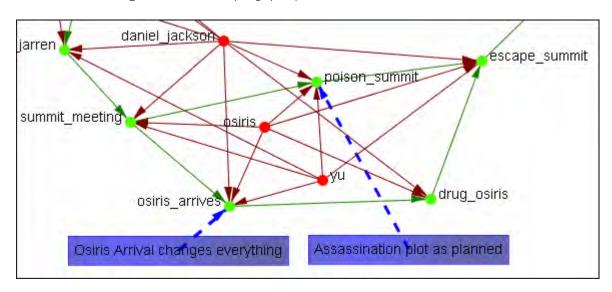
Reverse Direction : Reverses the **Source** and **Target** aspects of the link.

Delete Link : Removed the link from the Network.

Comments

Add Comment : Attaches a comment to the **currently selected node**. Good for making notes for display purposes.

Mouse Menus

Right clicking any node in the Visualizer brings up a menu of the most useful tasks concerning nodes.
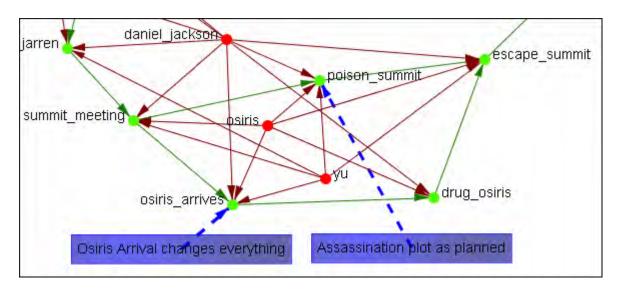
The Pin Node Submenu : Allows you access various function involving **pinning and unpinning** nodes in the visualizer

The URI Submenu : Allows you to attach a URI to a node or go to a URI already attached to a node.

The Appearance submenu : allows you to control the size of the individual node and font, brighten or darken it's color, or show or hide an individual label. You can even assign an individual picture to a node which is helpful when you are familiar with the agents, locations, or items.

MetaNodes : After selecting two or more nodes you can combine them into a single MetaNode and give that MetaNode a name. If necessary you can also Expand that MetaNode into its individual nodes.

Comments : You can add or delete comments for any node in the Meta-Network.
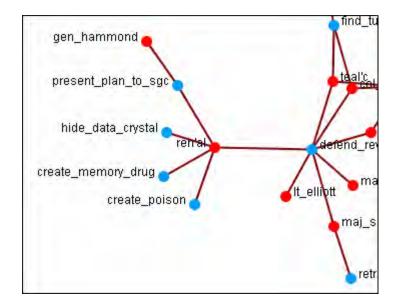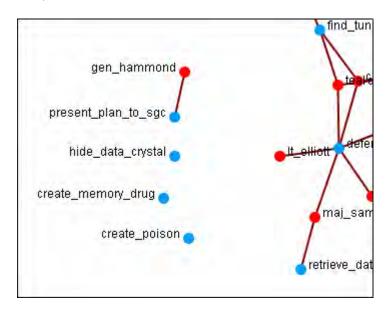
Removing Nodes In The Visualizer

You can remove nodes directly from the ORA Visualizer by right clicking on a node and selecting **Hide Node**. Below is an example of removing an node using the Hide Node tool.

Find the node you want to hide. Here we will hide **ren'al**.

Place the cursor over the node and **right-click** on the node to bring up the contextual menu. Highlight **Hide node**. **gen_hammond** and the four task nodes are now separated from the rest of the network.
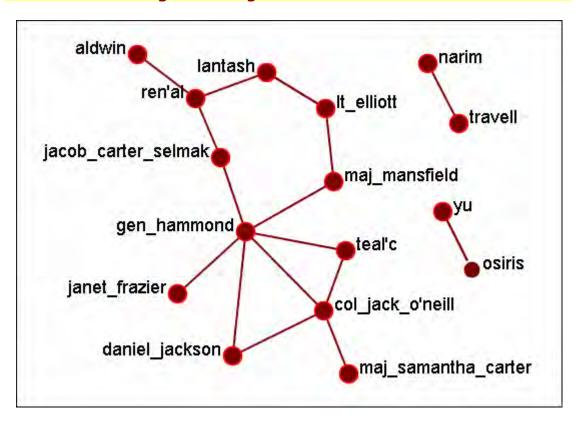
Set Node Picture

In addition to changing the size of nodes in the visualizer you can also assign nodes individual pictures. Again we start out with the stargate agent x agent nodeclass.
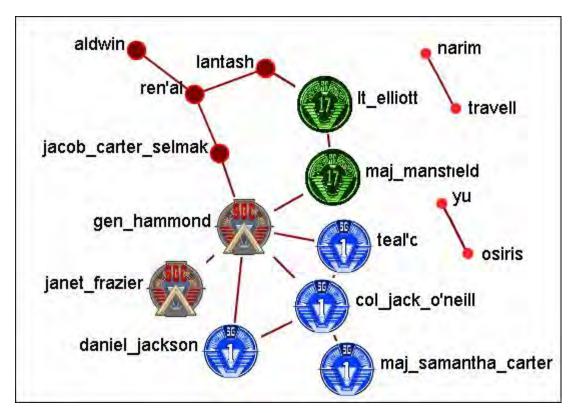
Right-Click on a node and select **Appearance Menu > Set Picture**. This brings up the Set Picture dialog. Click the downward triangle and select the **Agent - Stargate** folder. The Stargate pictures and symbols will appear.
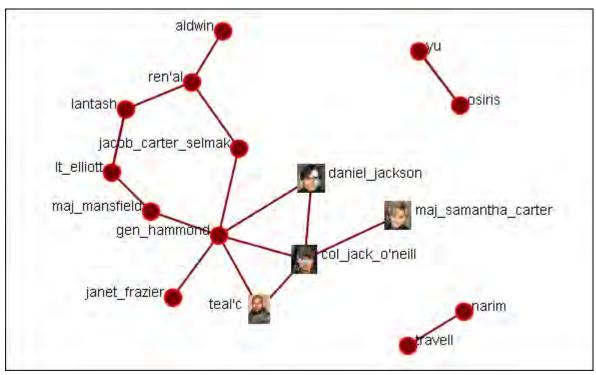
371

Symbol Replacement
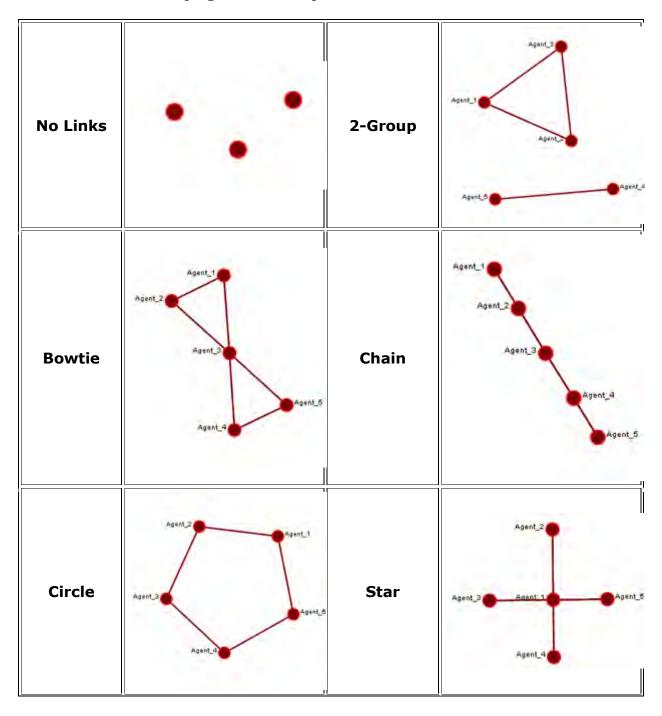
Here I'll use three different symbols to denote **1)** the SG-1 team (the blue circles a 1), **2)** the SG-17 team (the green circles with a 17), and **3)** the SGC who are not members of either team (the generic SGC symbol). You **Right-Click** on a node, select **Set Picture** and the Set Picture dialog appears. Select the picture folder to use then select the picture to use.
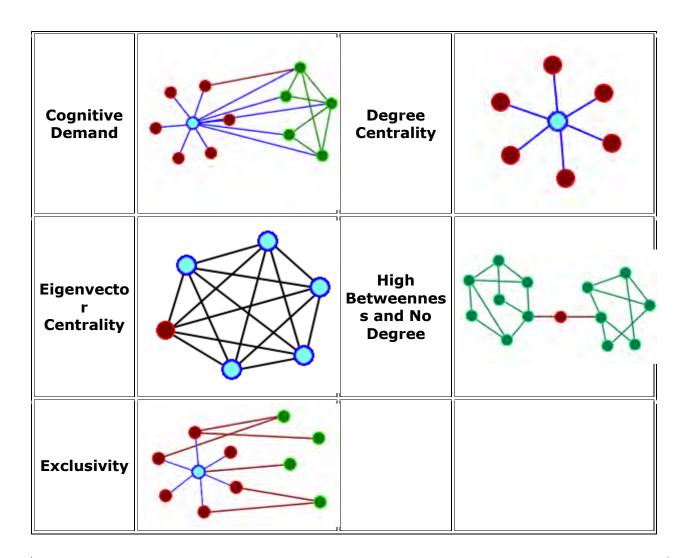
![Motifs icon] **Motifs**

This is a list of common motifs that can make it easier to find various clusters as the underlying structure of your network.

No Links		**2-Group**	
Bowtie		**Chain**	
Circle		**Star**	

Cognitive Demand		**Degree Centrality**	
Eigenvecto r Centrality		**High Betweennes s and No Degree**	
Exclusivity			
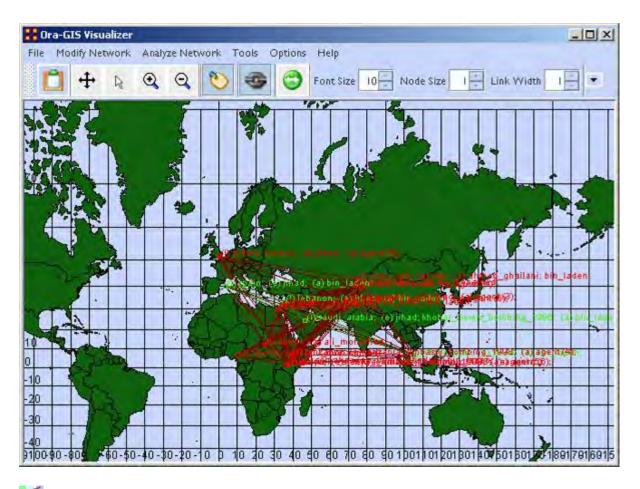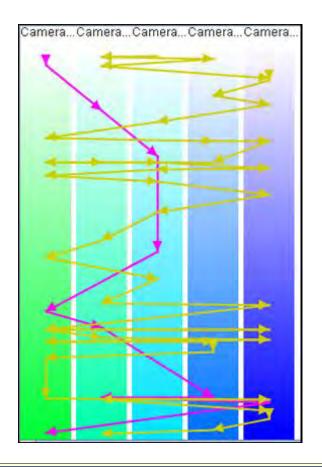
GeoSpatial Network

ORA provides tools to visualize Geospatial Information Systems: GIS Visualizer and GIS Visualizer 3D. When working with the GIS Visualizer your Meta-Network must include **location** information.

GeoSpatial is concerned with plotting who or what was where on a global basis. It plots nodes on a map and then connects those nodes.

![] **Loom** also deals with nodes in locations but is keyed on the timeframes in which nodes appear. There are locations in which two agents appear in but which were there at different times and never met.

376

![GIS icon] **GIS**

The GeoSpatial Visualizer differs from the 2D Visualizer as it places nodes on a representative map whereas the regular visualizer allows the nodes to float freely in a **non-space**.

Many **Geospatial information system (GIS)** products apply the term geospatial analysis in a very narrow context. In the case of vector-based GIS this typically means operations such as map overlay (combining two or more maps or layers according to predefined rules), simple buffering (identifying regions of a map within a specified distance of one or more features, such as towns, roads or rivers) and similar basic operations. The Geospatial Network tool in ORA allows for the plotting of a Meta-Network on a map of the globe using **longitude** and **latitude** for placing the nodes.

This **longitude** and **latitude** is associated with a **location** nodeclass. Then other nodeclasses are associated with the location nodeclass. This information is used to place people and items on the map.

Below is the default map included with ORA's GIS tool.

The ORA Geospatial Visualizer can be accessed from either the main menu bar: **Main Menu Bar > Visualization > Geospatial Networks** or by clicking the triangle on the Visualizer button in the Editor.

Geospatial Networks attributes

The Meta-Network **MUST** contain a **location Node Class** and that Node Set **MUST** contain **longitude** and **latitude** data. Below is an example of a **Location** node class information that contains Longitude and Latitude as attributes.

Location	COUNTRY	POP	CAP	LONGITUDE	LATITUDE
Murmansk	Russia	468000	N	33.0860405	68.96355
Arkhangelsk	Russia	416000	N	40.6461601	64.52067
Saint Petersburg	Russia	5825000	N	30.4533272	59.95189
Magadan	Russia	152000	N	150.780014	59.571
Perm'	Russia	1160000	N	56.2324638	58.00024

Yekaterinburg	Russia	1620000	N	60.6101303	56.84654
Nizhniy Novgorod	Russia	2025000	N	43.94067	56.28968
Glasgow	UK	1800000	N	-4.26994753	55.86281
Kazan'	Russia	1140000	N	49.1454659	55.73301

NOTE : The columns which are necessary to build a **Location Node Class** are highlighted in Blue. In this example below, COUNTRY, POP, and CAP are not necessary, though they can be added, as additional attributes. Further below this chart, we will display how a location node set will appear in ORA's editor.

Longitude and Latitude Format : The longitude and latitude use the **degree.minute** format. **Longitude** begins at the prime meridian which runs through Greenwich Village and run eastward 360 degrees. You can also denote the longitude values with positive or negative (-) numbers between 0 and 180. **Latitude** uses a 0-point which begins at the equator and runs to 90 to the north and -90 to the south.

Location data example

	Node ID	Node Title	ethno-reli...	friendship	latitude	longitude	political-le...
	<set filt... ▼	<set filt... ▼	<set filt... ▼	<set filt... ▼	<set filt... ▼	<set filt... ▼	<set filt... ▼
☐	afghanistan	afghanistan			34.28	69.11	
☐	africa	africa					
☐	airport	airport					
☐	britain	britain	christian	friend	51.36	0.05	liberal
☐	cape_town	cape_town					
☐	dar_es_sal...	dar_es_sal...					
☐	darfur	darfur		friend	12	24.45	
☐	egypt	egypt	sunni	friend	30.01	31.14	
☐	europe	europe		friend			liberal
☐	farm	farm		friend			
☐	indonesia	indonesia	sunni	friend	6.09	106.49	liberal
☐	israel	israel	jewish	friend	31.71	35.1	
☐	karachi	karachi					
☐	kenya	kenya		friend	1.17	36.48	
☐	lebanon	lebanon	sunni	friend	33.53	35.31	
☐	liberia	liberia			6.18	10.47	
☐	london	london	christian	friend	51.36	0.05	
☐	manhattan	manhattan	christian	friend	40.47	73.58	liberal

GeoSpatial Network Toolbar

The GUI contains tools to work with the GIS maps. You can also customize the toolbar with functions from the menu.

Standard Toolbar
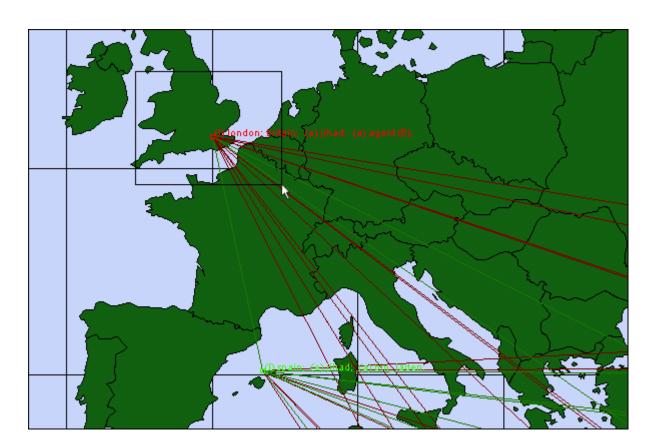
Copy Image to Clipboard : Places a copy of the current map in the clipboard which can be pasted into another program.
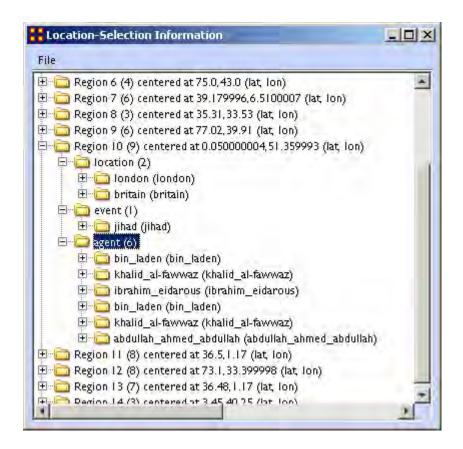
Pan Map : Place the cursor on the map and click the button. The map will re-center on those co-ordinates.

Select Place : Click and drag to capture nodes for information to display in an information window. This information will be displayed in an expanding tree format. Click and drag the cursor to include the nodes to display.

A pop-up window will display information for all the nodes captured by the cursor.

Zoom In : Zoom In works in two ways. 1) Click and drag to select the area you want to zoom in on. The selected area will fill the GIS window. 2) Click the map once and the view will zoom in centering on the clicked area. This can be repeated until you zoom in as close as needed.

Zoom Out : Place the cursor on the map and click the button. the map will zoom out and re-center on those co-ordinates.

Show Labels : Toggle the labels on & off.

Show Links : Toggle the links on & off.

Show Arrows : Toggle the arrows on & off.

Font Size : Increase or decrease font size using the up & down arrows.

Node Size : Increase or decrease node size using the up & down arrows. The size of the node is proportional to the number of entities in a location.

Link Width : Increase or decrease link width using the up & down arrows.

GeoSpatial File Menu

The GeoSpatial file menu allows you to open, add, and save map data.

Open Meta-Network... : Opens a **NEW** Meta-Network file into the Geospatial map. REMEMBER: The new file needs to have both **longitude and latitude** data.
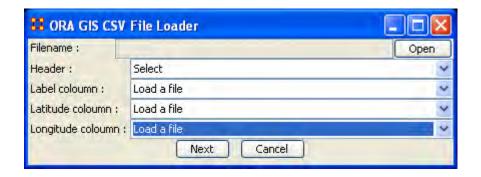
Open Trailset... :

Add GIS Data... :

Add CSV : You can load a **.csv** file and specify which columns contains the longitude/latitude data.

Add KML (Google Earth) : KML is a file format used to display geographic data in an earth browser (e.g. Google Earth). A KML file is processed in much the same way that HTML (and XML) files are processed by web browsers. Like HTML, KML has a tag-based structure with names and attributes used for specific display purposes.

Add e00 : Contains Geographic Information Systems (GIS) mapping data exported by ArcGIS software. ORA can import this style of vector data and convert it for use in the GeoSpatial tool.

Add ERSI Shapefile : An ERSI shapefile is a digital vector storage format for storing geometric location and associated attribute information. Shapefiles spatially describe geometries: points, polylines, and polygons.

NOTE : This format lacks the capacity to store topological information.
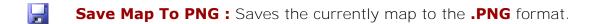
Add Shapefile (using Openmap) :
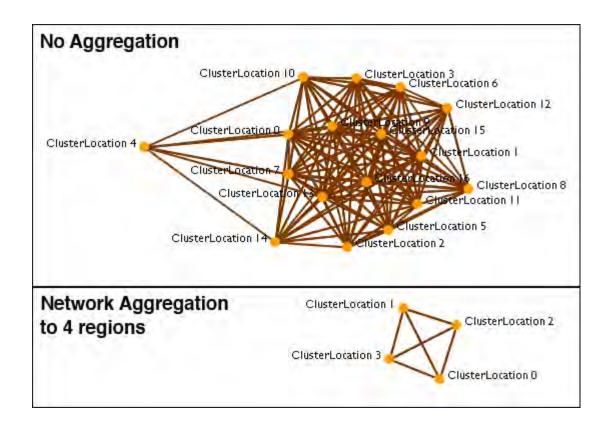
Add Shapefile (using Geotools) :

Save Map… :

Save Map To PNG : Saves the currently map to the **.PNG** format.

Save Map To SHP : Saves the currently map to the **.SHP** format.

Save Map To KML : Saves the currently map to the **.KML** format.

Save Map To DyNetML : Saves the current map to the **DyNetML** format with the added NodeClass **ClusterLocation--ClusterLocation**. This records changes affected by the Network Aggregator in grouping locations into regions.

Save Map To Loom Trailset : Saves the currently map to the **Loom** format.

Current GIS Layers

Open From Defaults : Opens what is currently saved as the default Dynamic Network and GIS layers.

Open from a File : Uses the information saved in the choose file to set the Dynamic Network and GIS layers.

Save As Default : Saves the currently selected Dynamic Network and GIS layers as the default.

Save to a File : Saves the currently selected Dynamic Network and GIS layers to a file.

Current Meta-Network Locations Configurations

Open From a File :

Save to a File :

Create Meta-Network From Visible Network...

Copy Image To Clipboard : Saves a picture of the current map to the clipboard which can be pasted into another program for use in demonstration purposes.

Close

GeoSpatial Modify Menu

GeoSpatial Actions Menu.

Replace Current Meta-Network : Allows you to open a new geospatial file without closing the GIS visualizer.

Create New Spatial Relation :

Create New Spatial Nodeset :

GeoSpatial Analyze Menu

GeoSpatial Actions Menu controls color and size of location nodes.

Reset Map Zoom : Reset the display of the map to it's default size.

Reset Nodes Colors and Size : Returns on nodes to their default colors and sizes.

Size Nodes by Attribute or Measure : Sizes nodes using numerical data.

Size Links by Value : Adjusts the weights of links in proportion to all links.

Color Nodes by Attribute or Measure : Colors the nodes using

Color Nodes by Component :

Color Nodes by Concor Grouping : Colors nodes using **Concor Grouping***.

Color Nodes by Newman Grouping : Colors nodes using **Newman Grouping***

GeoSpatial Tools Menu

Tools is a set of controls for adjusting how nodes and links are displayed.

Layer Manager : Controls the visibility of the NodeClass(es), Network(s), and Shape layers on the map by checking or unchecking the boxes.
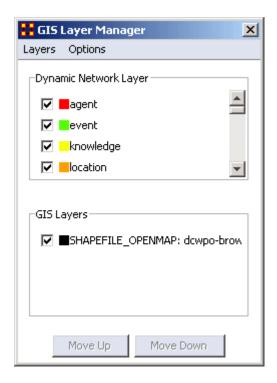
Network Aggregator : Controls the groupings of nodes which are displayed together as separate regions.

Get/Set View :

Configure Meta-Network Locations :

Layer Manager

The **GIS Layer Manager** controls what is displayed in the GeoSpatial Visualizer. This includes all NodeClasses and Shape layers. It is similar to the Legend in the regular Visualizer.

Selecting or Unselecting a NodeClass or GIS Layer will remove it from the display. It does not remove it from the dataset.

The **Move Up** and **Move Down** buttons adjust the display position of the SHAPE files. These buttons have no affect on the Network Layers.

Layer Manager Menus

Layers

   Add New Layer : You can add additional map files to the display.

   Remove Selected Layer : You can remove any selected map layer that has a checkmark next to it from the display.

   Show All Network Layers : Displays all node information on the map

Hide All Network Layers : Removes all Network information form the map. The Information is still available through the Layer Manager.

NOTE : If you have more than one map layer loaded you can arrange how they are stacked.

Change Layer Position > Move Layer Forward :

Change Layer Position > Move Layer Backward :

Change Layer Position > Move Layer To Front :

Change Layer Position > Move Layer To Back :

Add Layer As Nodeset :

Options
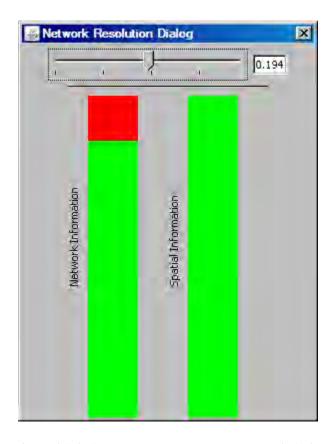
Set Current Map Data As Default :

Reset Map With Default Data :

Network Aggregator

The Network Aggregator looks at regions instead of individual locations. Using the slider you can control how dense each region will be. Increasing the density groups more locations into a region.
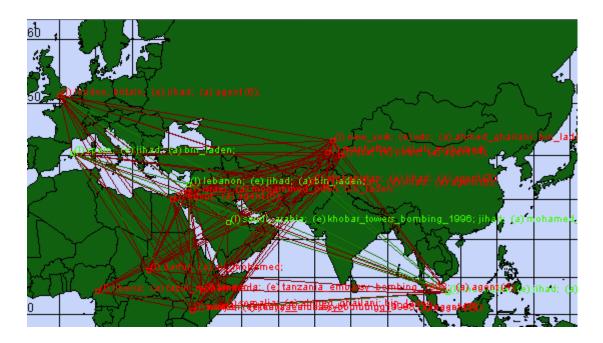
Using a sliding scale, which in turn represents network information retention (or loss) depending on the level of the adjusted slider. A value can also be entered manually in the numeric field box.

When the network bar is fully **red** it indicates full network resolution and therefore network information is preserved and displayed in the GIS Visualizer. Conversely, when lower network resolution figures are entered network detail can be sacrificed. This bar graph is intended to show the level of such loss.
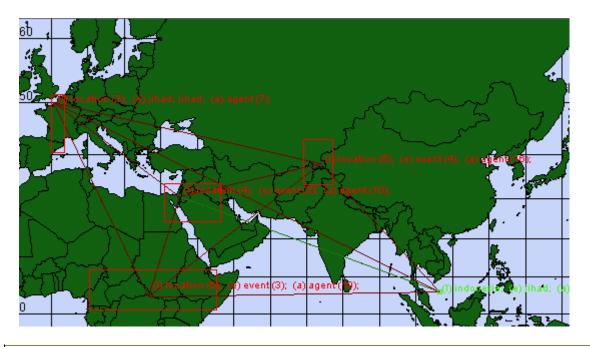
It is best to experiment with the slider at varying values and observing GIS visualization for optimum balance of data resolution and visual relevancy.

This is a useful tool when you have many noisy or cluttered locations. The grouping assists you in seeing the connections easier.

GeoSpatial with no aggregation

GeoSpatial using Network Aggregation
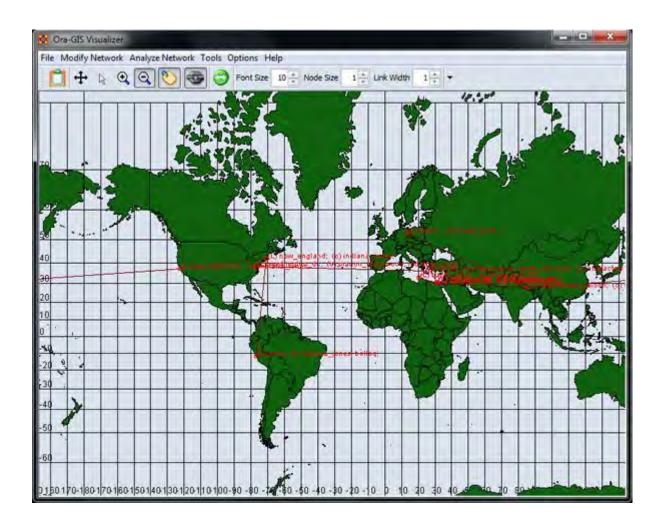
GeoSpatial Options Menu

These options control various display options of the Visualizer.

Show Links : Toggles the visibility of the Links.

Show Labels : Toggles the visibility of the Node Labels.

Show Arrows : Toggles the visibility of the link Arrows.

Use 2D Visualization (Openmap) : Switches to the 2D map view.

Use 3D Visualization (NASA Worldwind) : Switches no the 3D map view.

NASA World Wind is mapping software developed by NASA. The globe can be rotated, tilted, and zoomed in and out. Nodes and links are plotted on the globe with word balloons holding descriptions of the nodes.

	Loom/View Trails

Description

Loom is primarily concerned with how specific entities move or evolve through other entities. In the most basic case, this involves Agents as they move through Locations. More abstract examples include Agents moving through Beliefs or Beliefs moving through Locations.

When you first start Loom, you are prompted for a Network. The target nodes in the Network dictate what they're moving through. The source nodes are who/what is doing the moving.

Once started, Loom goes through all timeslices, and looks at the connections in that network only. It sets up a series of waypoints based on what connections are made at that timeslice, upon which it constructs a trailset.

Classically, we restrained Loom such that for any timeslice, any source node could appear in at most one target, but now we allow for a source node to occupy multiple targets at a time (Agents having multiple beliefs, for example).

NOTE : Loom can also display things like **persons through beliefs** showing how a person make change their mindset over time.

Before you Start

Before starting you need a dataset that can be used within the Loom. One such dataset is the TAVI dataset. Use **File > Open Meta Network** to open TAVI.xml
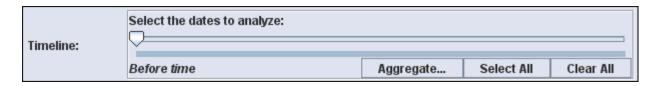
The clock icon in the Meta-Network Panel means that it is a dynamic meta network and that there are continuous-time updates **(deltas, which are listed below the top-level network)** indicating how the network changes over time.
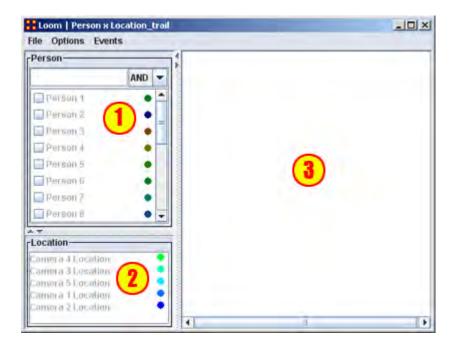
To analyze the data you must have a relation that **A)** changes over time and **B)** stays many:one (e.g., a person must have only one location at any given time). Usually this is Agent x Location, but in theory you can select other relations.

In the main ORA screen, open the **Visualizer > View Trails**. Select the **Person x Location** input network.

At the top left is a list of Agents; the bottom left is a list of locations. Activating the check-box next to an agent will cause its trail to appear in the right hand panel. The large vertical strips correspond to locations, and the path moving between them is the trail indicating the agent's location at each point in time.

Timeline:	Select the dates to analyze:			
	Before time	Aggregate...	Select All	Clear All
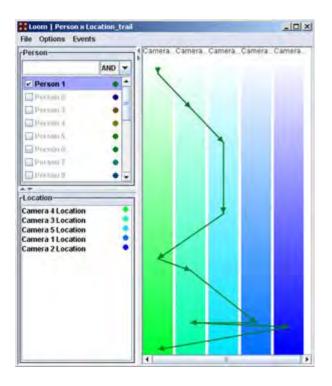
Open Loom : Highlight the TAVI dataset in the Meta-Network pane. From the drop down menu select **Visualizations > View Trailsets**. In the dialog box select **Persons x Locations**. The Loom window will appear with the chosen Network.
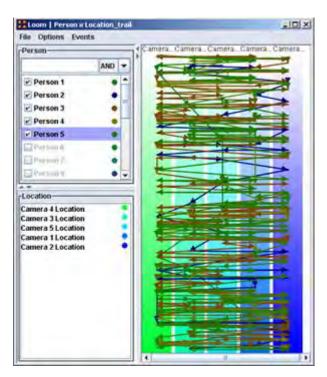
1. **Person Pane :** Lists all persons in the meta-networks. This pane contains a check box to the left of next to each person to control which persons are visible. To the right is a color designation

2. **Location Pane :** Lists the locations in the meta-networks. To the right is a color designation.

3. **Loom Pane :** Displays the persons as they travel through the locations.

Selecting one person you can easily see where they've been **(Left window)**. But when you select multiple people, it can quickly get impossible to view the trails **right window)**.

Making the Display more Readable : can be done in several ways. Clicking the color circle representing the individual node brings up the color changer. You can then select a color which stands out better for viewing.

 Loom File

 Loom File Menu

> **Open … :** Opens up DyNetML file which contains location and time data.

> **Save as… :** Saves currently open DyNetML file under a new filename.

> **Export > DyNetML :** Saves the Loom data as a DyNetML file.

> **Save Image as… :** Saves an image of the current Loom state. Can be saved as a **png, jpeg, pdf, svg, or tiff** file.
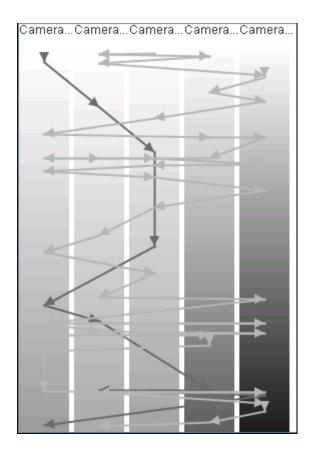
 Loom Options

 Loom Options Menu
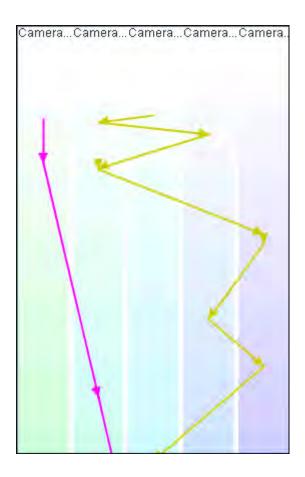
> **Anti-Aliasing :** Used to smooth out rough edged lines.

> **Grayscale :** Removes all colors from image and is good when printing black and white images.
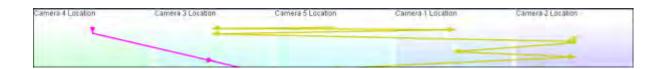
Show Timeline : Displays the timeline markers on the left of the Loom.

Stretch Vertical : You can select a percentage to enlarge the height of the Loom. Good when many nodes are situated in a small area.
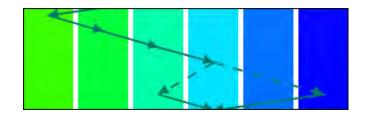
Bar Width : Widens the individual locations. Good when many nodes are situated in a small area.

Allow Multiple Waypoints per Period : If unchecked a node will appear **only** in the most weighted position. If checked then the node will be placed in all locations.
Example : A main location also contains three separate rooms. An agent can be listed to be in the main location but can also be listed as moving between the individual rooms. If this option is checked, the agent will be shown to be in all locations which can lead to it appearing an agent is in two places at once.

NOTE : The dotted line denotes a node which disappeared from all locations then reappeared later on.

Loom Clustering

What Trail Clustering can do:

Classification : Given a set of trails, the algorithm can classify trails based on the underlying patterns of observed location visits.

Level of Similarity : The degree of similarity in the classified trails can be increased or decreased according to user need by varying the specified depth of the PFSA.

Approximate similarity : Further, trails that have a subset of similar patterns of location visits, while also displaying differences by visiting other non common locations can also be identified by specifying a threshold for the similarity.
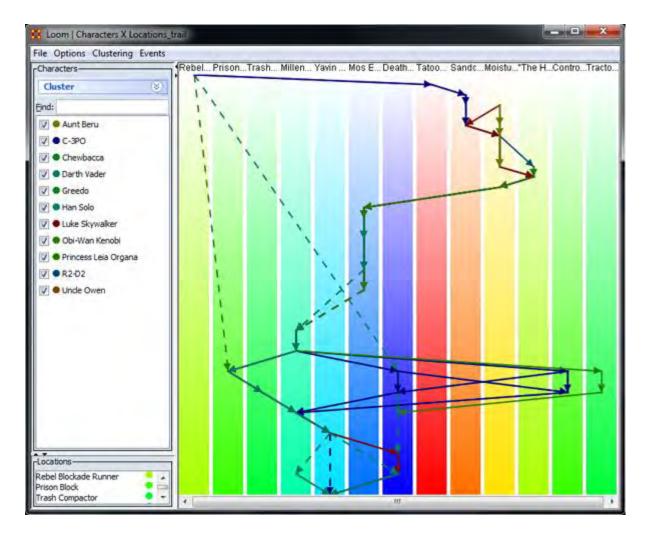
Noise Insensitivity : Approximate clustering also provides an inbuilt imperviousness to measurement noise in the trails. Occasional incorrectly identified location in a trail will not prevent it from being classified correctly according to the other correctly identified locations.

Pre-specification of number of clusters : The use of agglomerative clustering eliminates the need of pre-specifying the required number of clusters. Thus, this removes the risk of assigning trails to clusters they do not necessarily belong to (such as in cases where the number of final clusters is less than that necessitated by the dataset)

Creating a Cluster
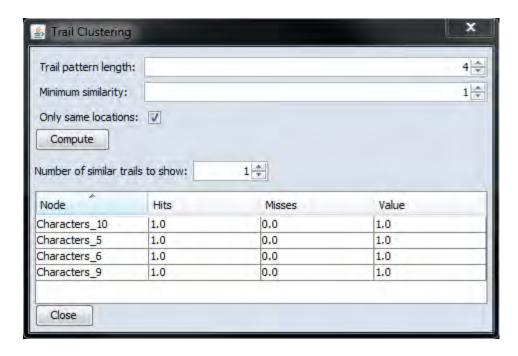
After loading a Loom ready DyNetML set (e.g. The Star Wars dataset) You can show how the characters are clustered. Below is the Loom with entire Star WArs characters set displayed. With this example we are interested in the Aunt Beru/Uncle Owen and Han Solo/Chewbacca pairings. Both pairs followed, for the most part, the similar path.

NOTE : The Loom lists characters in alphabetical order which may, or may not, correspond to the order of the characters in a dataset.
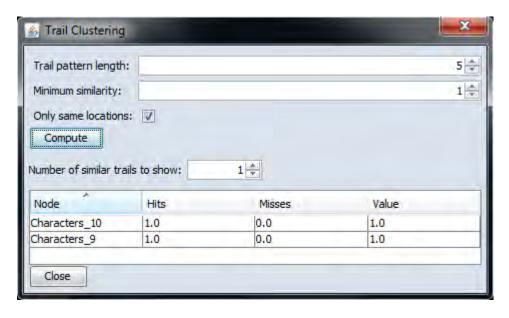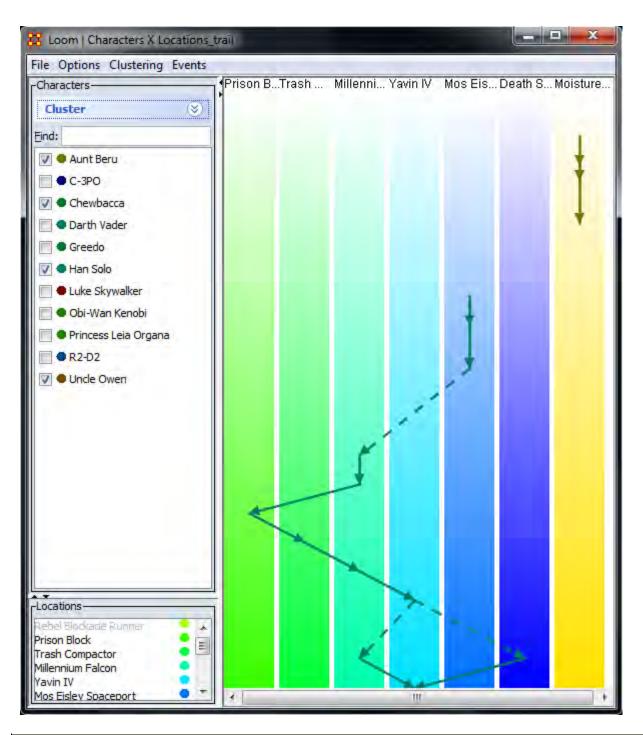
From the menu call up Trail Clustering by selecting **Clustering > Show Similar Trails**. Leaving most at their defaults I increase the **Trail pattern length** to four. Clicking [**Compute**] will display the four characters selecting in the Loom.

NOTE : In this example Uncle_Owen = character_5, Aunt_Beru = character_6, Han_Solo = character_9, and Chewbacca = character_10.

Increasing the Trail pattern length to five removes Uncle Owen (character_5) and Aunt Beru (character_6). If you look at the Loom image below both these nodes appear in only four time slices. When the length is increased to five they are removed form the display.

![Loom Events icon]	**Loom Events**

![Loom Events Menu icon] Loom Events Menu

Load Events from File : Load csv file containing two columns (date,events)

Group Events By... :

Don't Group : Makes each line correspond to one event.

Day : Each line will correspond to a summation of all events for a day

Month : Each line will correspond to a summation of all events for a month

Year : Each line will correspond to a summation of all events for a year

Show Events : A toggle to show/hide event lines.

Make Loom Dataset

Multiple Meta-Networks can be combined into a single Meta-Network usable in Loom.

Pre-Requisites
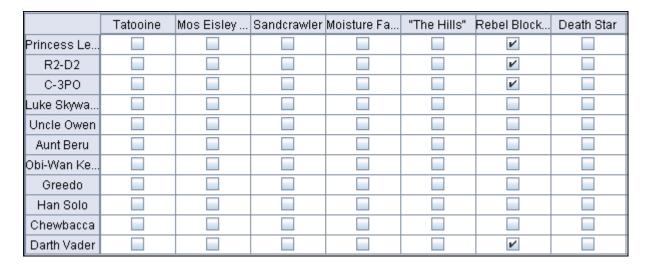
To create a dataset usable in Loom it requires multiple Meta-Networks containing the exact same Nodes/NodeClasses/Networks. What changes between the multiple Meta-Networks is which Nodes and Links are present in each timeslice.

NOTE : Every NodeClass needs to contain **identical nodes** in every NodeClass. **No exceptions**.

What differs is the Networks created from these agent and location NodeClasses. Below shows the difference between the first three time slices of the Star Wars set. Identical characters and locations but each network changes as to which characters are listed in individual locations.

Star Wars - Time Slice 1

	Tatooine	Mos Eisley ...	Sandcrawler	Moisture Fa...	"The Hills"	Rebel Block...	Death Star
Princess Le...	☐	☐	☐	☐	☐	✔	☐
R2-D2	☐	☐	☐	☐	☐	✔	☐
C-3PO	☐	☐	☐	☐	☐	✔	☐
Luke Skywa...	☐	☐	☐	☐	☐	☐	☐
Uncle Owen	☐	☐	☐	☐	☐	☐	☐
Aunt Beru	☐	☐	☐	☐	☐	☐	☐
Obi-Wan Ke...	☐	☐	☐	☐	☐	☐	☐
Greedo	☐	☐	☐	☐	☐	☐	☐
Han Solo	☐	☐	☐	☐	☐	☐	☐
Chewbacca	☐	☐	☐	☐	☐	☐	☐
Darth Vader	☐	☐	☐	☐	☐	✔	☐

Star Wars - Time Slice 2

	Tatooine	Mos Eisley ...	Sandcrawler	Moisture Fa...	"The Hills"	Rebel Block...	Death Star
Princess Le...	☐	☐	☐	☐	☐	☐	☐
R2-D2	☑	☐	☐	☐	☐	☐	☐
C-3PO	☑	☐	☐	☐	☐	☐	☐
Luke Skywa...	☐	☐	☐	☐	☐	☐	☐
Uncle Owen	☐	☐	☐	☐	☐	☐	☐
Aunt Beru	☐	☐	☐	☐	☐	☐	☐
Obi-Wan Ke...	☐	☐	☐	☐	☐	☐	☐
Greedo	☐	☐	☐	☐	☐	☐	☐
Han Solo	☐	☐	☐	☐	☐	☐	☐
Chewbacca	☐	☐	☐	☐	☐	☐	☐
Darth Vader	☐	☐	☐	☐	☐	☐	☐

Star Wars - Time Slice 3
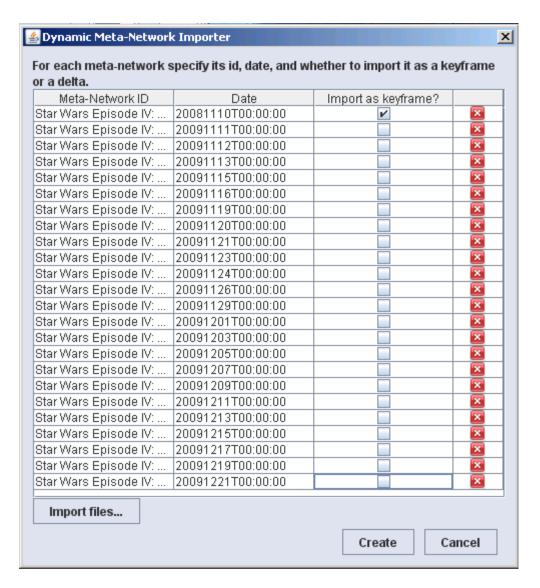
	Tatooine	Mos Eisley ...	Sandcrawler	Moisture Fa...	"The Hills"	Rebel Block...	Death Star
Princess Le...	☐	☐	☐	☐	☐	☐	☐
R2-D2	☐	☐	☑	☐	☐	☐	☐
C-3PO	☐	☐	☑	☐	☐	☐	☐
Luke Skywa...	☐	☐	☐	☐	☐	☐	☐
Uncle Owen	☐	☐	☐	☐	☐	☐	☐
Aunt Beru	☐	☐	☐	☐	☐	☐	☐
Obi-Wan Ke...	☐	☐	☐	☐	☐	☐	☐
Greedo	☐	☐	☐	☐	☐	☐	☐
Han Solo	☐	☐	☐	☐	☐	☐	☐
Chewbacca	☐	☐	☐	☐	☐	☐	☐
Darth Vader	☐	☐	☐	☐	☐	☐	☐

Creating a DataSet

Load all Meta-Networks into ORA. Highlight all these Meta-Networks.
Right-Click to bring up the contextual menu.

Select Create Dynamic Meta-Network. This brings up the Create dialog box.

Clear check marks under *Import as keyframe* from all instance except the first. Then click the **Create** button.

Creates Meta-Network with Clock

Now you have a Dynamic Meta-Network that can be viewed in Loom.

Deltas and Keyframes

`
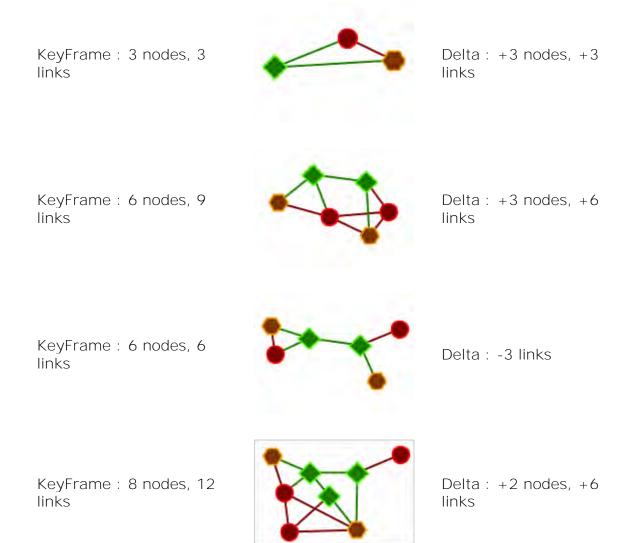
The difference between Deltas and KeyFrames

The **Keyframe** acts as a starting point, or a snapshot of what the meta-network looks like at any given time. It doesn't care what came before or will occur after.

The **Delta** is a set of instructions on how to change the meta-network. It reviews what came before it, applies a set of changes, and displays the revised meta-network.

Initial State : Nothing   **Initial State : Nothing**

KeyFrame : 3 nodes, 3 links

Delta : +3 nodes, +3 links

KeyFrame : 6 nodes, 9 links

Delta : +3 nodes, +6 links

KeyFrame : 6 nodes, 6 links

Delta : -3 links

KeyFrame : 8 nodes, 12 links

Delta : +2 nodes, +6 links

The biggest advantage of Deltas is it requires less information than KeyFrames. KeyFrames are a screenshot of every node and like in every time slice. Deltas, on the other hand, require only the information about individual changes.

NOTE : Change in the first KeyFrame do not affect proceeding KeyFrames. But with Deltas changes in the initial KeyFrame are taken into account by each Delta.

Importing Deltakey Frame Data

The easiest way to create a dynamic meta-network is to first create a number of regular meta-networks. Next, import all these neta-networks into ORA. Select all the meta-networks in the meta-network pane you want to merge and right-click. From the contextual menu select **Create a**

410

Dynamic Meta-Network. This brings up a Dynamic Meta-Network importer dialog with all your selected meta-networks.

Next you need to select which meta-networks you want to act as keyframes. This can be as few as the first meta-network or as many as all. Then click the [**Import**] button. You have now created a Dynamic Meta-Network.

Delta KeyFrame and Loom

The question arise as to whether you need a Delta KeyFrame to run this in Loom. No, a Delta KeyFrame Meta-Newwork in not a necessity for Loom as long as the locations/nodeclass/node id's match in every Meta-Network.

How can you show only some time periods or some locations

In Loom, only the locations that have entities currently moving through them are displayed. Locations with entities moving through them **cannot be hidden**.

Similarly, there's no way to currently turn off individual time periods. The work-around would be to just not include that time-periods meta-network whenever you load up loom.

ShapeFiles

`

Default

The default GIS renderer is known as OpenMap. Through OpenMap you can aggregate locations, add shapefiles, and perform basic visualization.
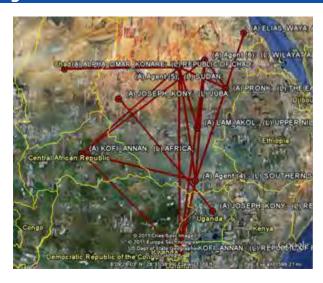
NASA WorldWind

Once the data has been loaded into GIS, under options you can select the 3D visualization, which uses Nasa Worldwind. Any feature available in the 2D version is available in the 3D version, albeit it may operate a tad differently 3D Version operates primarily as a sphere, but a flat map version can be toggled on.

Google Earth

Google Earth is not integrated with ORA, but the GIS is capable of producing a **.kml file** for use with Google Earth. The command to do so is located under File -> Save Map. The output will have the same nodes and links.

Customizing

Shapefiles allow you to customize the map with custom political boundaries, highways, whatever you can find Shapefiles are imported via the File -> Add GIS Data. menu. Both the fill color and border color can be changed via the GIS Legend. All shapefiles must be in WGS84 Projection Screenshot shows a shapefile of the political boundaries of the states of Sudan, as downloaded from MapLibrary.org

MGRS Coordinates

MGRS coordinates consist of an even number of characters. It starts off identifying a Grid Zone, and then as more characters are added the location becomes more precise.

To implement, create an Attribute called the **Military Grid Reference System** [**MGRS**] in your location node class. Unlike Lat/Lon, it only requires a single attribute. Populate each location with it's corresponding MGRS, and so on. Apart from it's separate conventions, it operates much the same as the Lat/Lon, in terms of adding it to your data.

NOTE : In terms of Priorities, GIS looks for Lat/Lon data first, then attempts to find MGRS.
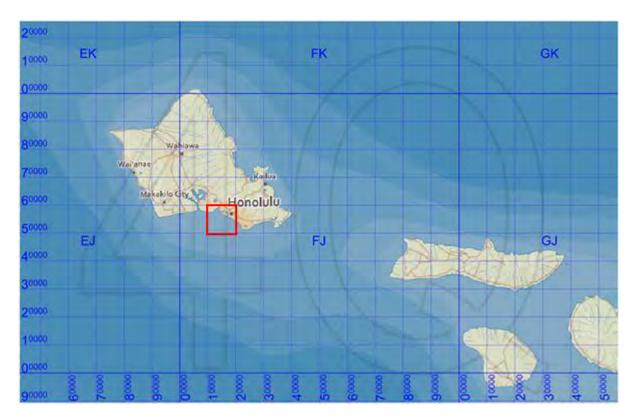
Example

An example of an MGRS coordinate, or grid reference, would be **4QFJ12345678**, which consists of three parts:

- 4Q (grid zone designator, GZD),

- FJ (the 100,000-meter square identifier), and

- 12345678 (numerical location; easting is **1**234 and northing is **5**678, in this case specifying a location with 10m resolution). These map coordinates are read from west to east first (easting), then from south to north (northing).

4QFJ15 is the square occupied by Honolulu. The **1 and 5** are the first digits of each of the four-digit combinations. The rest of the digits are used as refinement to the location.

ORA Measures

ORA contains over **100 measures**.

What is a Measure

A measure is a function that takes as input a Meta-Network and outputs a single value or a vector of values. Consider the measure **Density**. The output for this measure is a single number used to analyze an organization. By default, all measures are run on a Meta-Network. To view which measures are available, go to the Tools menu and select the **Measures Manager**.

ORA Measures Manager

The Measures Manager categorizes measures in the following ways: **Entity Level, Graph Level, and Risk Category.**

Entity Level Measure is one that is defined for, and gives a value for, each entity in a network. If there are x nodes in a network, then the metric is calculated x once each for each node.

Examples are Degree Centrality, Betweenness, and Cognitive Demand.

Graph Level Measure is one that is defined for, and gives a value for, the network as a whole. The metric is calculated once for the network.

Examples are Centralization, Graph Hierarchy, and the maximum or average Betweenness

*Some reports use a predefined set of measures, and these are not affected by the Measure Manager selections **(for example, the Intel, Context, Located SubGroups, Sphere of Influence,** and **Immediate Impact** reports). The **Risk Report**, however, uses only the measures selected in the Measure Manager.*

*The following sets of entities (with their abbreviated symbol) are used throughout the document: Agent (**A**), Knowledge (**K**), Resource (**R**), and Task (**T**). The following networks defined on these entity sets are used throughout the documentation.*

Symbol	Entity Sets		Name
	U	**V**	
AA	Agent	Agent	Communication Network
AK	**Agent**	**Knowledge**	**Knowledge Network**
AR	Agent	Resource	Capabilities Network
AT	**Agent**	**Task**	**Assignment Network**
KK	Knowledge	Knowledge	Information Network

KR	Knowledge	Resource	Training Network
KT	Knowledge	Task	Knowledge Requirement Network
RR	Resource	Resource	Resource Substitute Network
RT	Resource	Task	Resource Requirement Network
TT	Task	Task	Precedence Network

Input:

*This lists each of the Networks that are required as input to the measure. If a measure takes as input a specific network from the Meta-Network, such as the **Agent x Knowledge (AK) matrix**, then it is listed. If the measure runs on any **square (unimodal) network then N is given**. A measure can require more than one input matrix.*

1. *Each input matrix can have one or more of the following requirements:*

    a. ***square:*** *the matrix must have the same number of rows as columns*

    b. ***binary:*** *it must be binary data (i.e. any links in the network are assumed to have a weight of one)*

    c. ***symmetric:*** *the network must be undirected*

2. *ORA does the following when a network does not meet one or more of the above requirements:*

    a. ***square:*** *measure is not computed*

    b. ***binary:*** *measure is computed, and edge weights are ignored (all edges are given value one)*

    c. ***symmetric:*** *measure is computed, and the matrix is first symmetrized using the union method*

A complete list of all measures available in ORA, along with references, input and output specifications, can be found in the following ORA Measures sections.

Access Redundancy

Go To Measures by Report

Informal Name

None

Citation

Carley, Kathleen M. 2002. "Summary of Key Network Measures for Characterizing Organizational Architectures." Unpublished Document: CMU 2002

Minimum and Maximum Values

Minimum : 0

Maximum : $\Re \in [0, (|A|-1*|R|]$.

Computational Time

Fast

Description

Average number of redundant agents per resource. An agent is redundant if there is already an agent that has access to the resource.

Input :

Agent by Resource : DataType=binary.

Output :

Graph Level : DataType=real.

$$\Re \in [0, (|A| - 1) * |R|]$$

This is the Column Redundancy of matrix AR.

Actual Workload

Informal Name

None

Citation

Carley, Kathleen M. 2002. "Summary of Key Network Measures for Characterizing Organizational Architectures." Unpublished Document: CMU 2002

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

The knowledge and resources an agent uses to perform the tasks to which it is assigned. Individuals or organizations that are high in workload are those that are doing more complex tasks and have the resources and knowledge or expertise to do those tasks. Tasks are more complex if they require more expertise and/or more resources. This measures is only calculated if all of the following matrices are available: agent x knowledge, agent x resource, agent x task, knowledge x task, resource x task.

Example : In a given organization, employees have access to certain knowledge (e.g., fix computers) and resources (e.g., computer parts). This measure would compute a value to reflect the overall level of organizational access within the network.

Input :

Agent by Knowledge : DataType=binary

Agent by Resource : DataType=binary

Agent by Task : DataType=binary

Knowledge by Task : DataType=binary

Resource by Task : DataType=binary

Output :

Node Level : Type=agent and DataType=real.

$$\Re \in [0,1]$$

Actual Workload for agent i is defined as follows:

$$[AK * KT * AT'](i,i) / sum(KT)$$

Agent Knowledge Needs Congruence

Informal Name

None

Citation

Lee, 2004

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

The number of knowledge that an agent lacks to complete its assigned tasks expressed as a percentage of the total knowledge required for the assigned tasks.

Input :

Agent by Knowledge : DataType=binary.

Agent by Task : DataType=binary.

Knowledge by Task : DataType=binary.

Output :

Node Level : Type=agent and DataType=real.

$$\Re \in [0,1]$$

Agent Knowledge Needs compares the knowledge needs of the agent to do its assigned tasks, with the actual knowledge of the agent.

Let **N = AT*KT' = knowledge** needed by agents to do assigned tasks.

for Resource replace KT with RT

We need to sum the knowledge needed but not available.

Then, Agent Knowledge Needs for

$$\text{agent } i = \frac{\sum_{j=1}^{|K|} N_{i,j} *(\sim AK_{i,j})}{\sum_{j=1}^{|R|} N_{i,j}}$$

Agent Resource Needs is analogous, replacing AK with AR, and KT with RT.

Agent Knowledge Waste Congruence

Go To Measures by Report

Informal Name

None

Citation

Lee, 2004

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

The number of knowledge that an agent has that are not needed by any of its tasks expressed as a percentage of the total knowledge of the agent.

Input :

Agent by Knowledge : DataType=binary.

Agent by Task : DataType=binary.

Knowledge by Task : DataType=binary.

Output :

Node Level : Type=agent and DataType=real.

$$\Re \in [0,1]$$

Agent Knowledge Waste compares the knowledge of the agent with the knowledge it actually needs to do its tasks. Any unused knowledge is considered wasted.

Let N = AT*KT' = knowledge needed by an agent to do its assigned tasks.

for Resource replace KT with RT

We need to sum the knowledge the agent has but which is not needed.

Then, Agent Knowledge Waste for

$$\text{agent } i = \frac{\displaystyle\sum_{j=1}^{|K|} AK_{i,j} * (N_{i,j} = 0)}{\displaystyle\sum_{j=1}^{|R|} AK_{i,j}}$$

The equation for Agent Resource Waste replaces AK with AR, and KT with RT.

Agent Resource Needs Congruence

Go To Measures by Report

Informal Name

None

Citation

Lee, 2004

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

The number of resources that an agent lacks to complete its assigned tasks expressed as a percentage of the total resources required for the assigned tasks.

Input :

Agent by Resource : DataType=binary.

Agent by Task : DataType=binary.

Resource by Task : DataType=binary.

Output :

Node Level : Type=agent and DataType=real.

$$\Re \in [0,1]$$

Agent Knowledge Needs compares the knowledge needs of the agent to do its assigned tasks, with the actual knowledge of the agent.

Let **N = AT*KT' = knowledge** needed by agents to do assigned tasks.

for Resource replace KT with RT

We need to sum the knowledge needed but not available.

Then, Agent Knowledge Needs for

$$\text{agent i} = \frac{\sum_{j=1}^{|K|} N_{i,j} * (\sim AR_{i,j})}{\sum_{j=1}^{|R|} N_{i,j}}$$

Agent Resource Needs is analogous, replacing AK with AR, and KT with RT.

Agent Resource Waste Congruence

Go To Measures by Report

Informal Name

None

Citation

Lee, 2004

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

The number of resources that an agent has that are not needed by any of its tasks expressed as a percentage of the total resources of the agent.

Input :

Agent by Resource : DataType=binary.

Agent by Task : DataType=binary.

Resource by Task : DataType=binary.

Output :

Node Level : Type=agent and DataType=real.

$$\Re \in [0,1]$$

Agent Knowledge Waste compares the knowledge of the agent with the knowledge it actually needs to do its tasks. Any unused knowledge is considered wasted.

Let N = AT*KT' = knowledge needed by an agent to do its assigned tasks.

for Resource replace KT with RT

We need to sum the knowledge the agent has but which is not needed.

Then, Agent Knowledge Waste for

$$\text{agent } i = \frac{\displaystyle\sum_{j=1}^{|K|} AR_{i,j}*(N_{i,j}= 0)}{\displaystyle\sum_{j=1}^{|R|} AR_{i,j}}$$

The equation for Agent Resource Waste replaces AK with AR, and KT with RT.

Agent Socio Economic Power

Go To Measures by Report

Informal Name

None

Citation

Kathleen M. Carley, 2004

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Fast

Description

A measure of actor power based on access to knowledge, resources, and tasks in the organization.

Example : An individual within an organization that has the most connections, hence access, to knowledge (special skills), resources (unlimited funding) and tasks (has the most important tasks). Such an agent would rank high in the SocioEconomic Power measure.

Input :

Agent by Knowledge : DataType=binary.

Agent by Resource : DataType=binary.

Agent by Task : DataType=binary.

Output :

Node Level : Type=agent and DataType=real.

$$\Re \in [0,1]$$

Let N = [AK AR AT], that is, the concatenation of the input matrices.

Then, Agent Socio Economic Power is the row degree centrality of N.

Assignment Redundancy

Go To Measures by Report

Informal Name

None

Citation

Carley, Kathleen M. 2002. Summary of Key Network Measures for Characterizing Organizational Architectures. Unpublished Document: CMU 2002

Minimum and Maximum Values

Minimum : 0

Maximum : $\Re \in [0,(|A|-1*|T|]$

Computational Time

Fast

Description

Average number of redundant agents assigned to tasks. An agent is redundant if there is already an agent assigned to the task.

Input :

Agent by Task : DataType=binary

Output :

Graph Level : DataType=real

$$\Re \in [0, (|A| - 1) * T]$$

This is the Column Redundancy of matrix AT.

Authority Centrality

Informal Name

None

Citation

Kleinberg J.M, 1998

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

A node is authority-central to the extent that its in-links are from nodes that have many out-links. Individuals or organizations that act as authorities are receiving information from a wide range of others each of whom sends information to a large number of others. Technically, an agent

is authority-central if its in-links are from agents that have are sending links to many others. The scientific name of this measure is authority centrality and it is calculated on agent by agent matrices.

Input :

Square matrix : DataType=real.

Output :

Node Level : DataType=real.

Average CommunicationSpeed

Informal Name

None

Citation

Carley, Kathleen M. 2002. Summary of Key Network Measures for Characterizing Organizational Architectures. Unpublished Document: CMU 2002

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

The average speed with which any two nodes can interact. This is based on the inverse of the shortest path lengths between node pairs.

Example : In an organization that any two individuals have an average of 3.4 connections between them would approximate the average speed at

which an a communication might spread throughout the organization. Naturally, the lower the average links, the higher the organizational speed.

Input :

Square matrix : DataType=binary.

Output :

Graph Level : DataType=real.

$$\Re \in [0,1]$$

Average Speed is the average of the Inverse Closeness Centrality for the network entities.

Betweenness Centrality

Informal Name

None

Citation

Freeman, 1979

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Slow

Description

The Betweenness Centrality of node v in a network is defined as: across all node pairs that have a shortest path containing v, the percentage that pass through v. Individuals or organizations that are potentially influential are positioned to broker connections between groups and to bring to bear the

influence of one group on another or serve as a **gatekeeper between groups**. This agent occurs on many of the shortest paths between other agents. The scientific name of this measure is betweenness centrality and it is calculated on agent by agent matrices.

Example : This measure indicates the extent that an individual is a broker of indirect connections among all others in a network. Someone with high Betweenness could be thought of as a gatekeeper of information flow. People that occur on many shortest paths among other People have highest Betweenness value. Betweenness is one of the key measures used by those interested in networks.

Input :

Square Matrix : DataType=binary

Output :

Node Level : DataType=real

$$\Re \in [0,1]$$

Let G=(V,E) be the graph representation for the network. Let **n=|V|**, and fix an entity $v \in V$.

For $(u,w) \in V x V$, let $n_G(u,w)$ be the number of geodesics in G from u to w.

If $(u,w) \in E$, then set $n_G(u,w)=1$.

Define the following:

let

$$S = \{(u,w) \in VxV \ | d_G(u,w) = d_G(u,v) + d_G(v,w)\}$$

$$between = \sum_{(u,w)\in S}(n_G(u,v) * n_G(v,w)) / n_G(u,w)$$

let

Then **Betweenness Centrality** of the entity
$$v = between / ((n-1)(n-2)/2)$$

NOTE : If G is not symmetric, then between is normalized by (n-1)(n-2).

430

Betweenness Centrality Spatial

Informal Name

None

Citation

Olson, Mallory

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Slow

Description

The extent to which a node lies between other nodes that are far from each other.

Input :

Agent by Agent : DataType=real.

Agent by Location : DataType=real.

Output :

Node Level : Type=agent and DataType=real

Betweenness Network Centralization

Informal Name

None

Citation

Freeman, 1979

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Slow

Description

Network centralization based on the betweenness score for each node in a square network. This measure is defined for directed and undirected networks.

Input :

Square Matrix : DataType=binary

Output :

Node Level : DataType=real

$$\Re \in [0,1]$$

Let G=(V,E) represent the square network, and let n = |V|

let d_i = Betweenness Centrality of node i

let d = max{ d_i | 1 ≤ i ≤ n }

Then Network Betweenness Cent. = $\left(\sum_{1 \le i \le n} \overline{d} - d_i \right) / (n-1)$.

Biological Diversity Index

Informal Name

None

Citation

Minimum and Maximum Values

   Minimum : 0

   Maximum : 1

Computational Time

Fast

Description

The mathematical knowledge that helps is that the log of a product is the sum of the logs, e.g. **ln(4!) = ln(1)+ln(2)+ln(3)+ln(4)**. Otherwise it is impossible to calculate the index for larger data. So, the implementation is just simple loops:

```
float brillouin=0;

for (int i=1; i<=N; i++)

    brillouin = brillouin + log(i);



for (int s=1; s<=number_of_groups; s++)

    for (int i=1; i<=n[s]; i++)

        brillouin = brillouin - log(i);

brillouin = brillouin / N
```

[ln(N!) - Σ ln(ni!)] / N where ni is the number of elements in a specific group and N is the sum of all ni.

An issue of the Brillouin index is the lack of normalization, so there is no defined value range. The idea is that larger environments are more diverse even if the distribution of the elements is the same. Nevertheless the results look always similar because of the /N.

Input :

Node Level : DataType=Binary

Output :

A single value showing distribution.

Bonacich Power Centrality

Go To Measures by Report

Informal Name

None

Citation

Bonacich P., 1987 "Factoring and weighing approaches to clique identification" Journal of Mathematical Sociology, 2 (January, 1972), 113-120.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

The Bonacich Power Centrality computes the centrality of each entity based on the centrality of its neighbors. Beta should be chosen such that its absolute value is less than the reciprocal of the larges eigenvalue of N.

Example : In a given organization, this meausre can tell us who is connected to the most powerful (e.g., other highly connected agents) people.

Input :

Square Matrix : DataType=real and Directed=No.

$$\text{beta} \in \Re$$

Output :

Node Level : DataType=real.

$$\Re \in [0,1]$$

Let A = adjacency representation of the network N. Then Bonacich Power Centrality for the network is the solution c to the following system of linear equations:

$$(I - \beta A)^{-1} c = Ae$$, where I is the identity matrix, and e a vector of ones.

NOTE : If beta is 0, then c is simply equal to the row degree of each entity.

The values of c are not normalized.

Burt Constraint

Informal Name

None

Citation

Burt, Ronald. Structural Holes: The Social Structures of Competition. Cambridge, MA: Harvard University Press, 1992.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

The degree to which each node in a square network is constrained from acting because of its existing links to other nodes.

Example : Bob is given a certain task, but that task requires certain links, which Bob does not have. Bob is constrained in the sense he does not have the links to complete the task. This measure computes a value for such relationships as they distributed across the whole network.

Input :

Square matrix : DataType=real.

Output :

Node Level : DataType=real.

$$\Re \in [0,1]$$

This is the Constraint measure described by Equ. 2.4 on pg. 55 of Burt, 1992.

NOTE : The matrix Z is the adjacency matrix representation of the network N.

Burt Effective Network Size

Go To Measures by Report

Informal Name

None

Citation

Burt, Ronald. Structural Holes: The Social Structures of Competition. Cambridge, MA: Harvard University Press, 1992.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Fast

Description

The effective size of a node's ego network based on redundancy of links.

Example : In a given organizaton, let us say Bob has ties to many others in the organization, who in turn have redundant ties to other people. The point at which this no longer applies tends to define effective network size.

Input :

Square matrix : DataType=real.

Output :

Node Level : DataType=real.

$$\Re \in [0,1]$$

This is the Effective Size of Network measure described by Equ. 2.2 on pg. 52 of Burt, 1992. Note that the Communciation Network is used for the matrix Z.

Capability

Go To Measures by Report

Informal Name

None

Citation

Carley, Kathleen M., 2007

Minimum and Maximum Values

Minimum :

Maximum :

Computational Time

Fast

Description

Detects entities with high or low degree relative to other entities. The formula discounts for the fact that most agents have some connections and assumes that there is a general discount to having large numbers of connections.

Input :

Square matrix : DataType=real.

Output :

Node Level : Type=row and DataType=real.

$$\Re \in [0,1]$$

Let RD = the vector of row degrees for each entity in the network. Scale RD by dividing each element by the largest value in RD.

Then the Capability of node i is: $\dfrac{1}{\left(1.0 + e^{-(RD_i - 0.5)*10}\right)}$

NOTE : For the most connected entity, this value tends to 1.0, for the least connected node it tends to 0, and for the average connected entity it tends to 0.5.

Characteristic Path Length

Informal Name

None

Citation

NetStat

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

The average shortest path length between nodes, excluding infinite distances.

Example : In a given organization, this measure computes the average amount of connections between any two members in the organization.

Input :

Square matrix : DataType=binary

Output :

Graph Level : DataType=real.

$$\Re \in [0, N]$$

Let G=(V,E) represent a square network. Define a set S of all pairs (i,j) of entities such that i can reach j. Then average the shortest paths.

let **S = {(i,j) | j is reachable in G from j }**

Then,

$$\text{Average Distance} = \frac{\sum_{(i,j)\in S} d_G(i, j)}{|S|}$$

Clique Count

Go To Measures by Report

Informal Name

None

Citation

Wasserman, Stanley and Katherine Faust. "Social Network Analysis: Methods and Applications." Cambridge: Cambridge University Press, 1994.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Slow

Description

The number of distinct cliques to which each node belongs. Individuals or organizations who are high in number of cliques are those that belong to a large number of distinct cliques. A clique is defined as a group of three or more actors that have many connections to each other and relatively fewer connections to those in other groups. The scientific name of this measure is clique count and it is calculated on the agent by agent matrices.

Input :

Square matrix : DataType=binary.

Output :

Node Level : DataType=integer.

$$Z \text{ (not scaled)}$$

A clique of an undirected, square network is defined in Wasserman and Faust as a maximal complete subgraph of three or more nodes. The clique detection algorithm employed is the Bron-Kerbosch maximal clique algorithm, which is Algorithm 457 of the Collected Algorithms from CACM.

Closeness Centrality

Informal Name

None

Citation

Freeman, L.C. (1979). "Centrality in Social Networks I: Conceptual Clarification." Social Networks, 1, 215-239.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Slow

Description

The average closeness of a node to the other nodes in a network. Closeness is the inverse of the average distance in the network between the node and all other nodes.

Closeness reveals how long it takes information to spread from one individual to others in the network. High scoring individuals in Closeness have the shortest paths to all others in the network. It would follow such individuals could monitor the information flow in an organization better than most others that have a lesser Closeness value. The ones with the

highest value in this measure will often times have the best picture of what is happening in the network as a whole.

In short, Closeness is the inverse of the average distance in the network between any given node and all other nodes.

NOTE : This measure uses both link weights and link direction in its calculations.

Example : Closeness reveals so-called people in the know. Closeness measures the path length from one person to another in a network. It tells us which person is central to an organization.

Input :

Square matrix : DataType=real.

Output :

Node Level : DataType=real.

Let **G=(V,E)** be the graph representation of the square network.

Fix

$$v \in V$$

let

$$dist = \sum_{i \in V} d_G(v,i) \quad \text{with } d_G(v,i) = |V|$$

if i is unreachable from v.

Then Closeness Centrality of node $v = (|V|-1)/dist$.

Closeness Centrality Spatial

Go To Measures by Report

Informal Name

None

Citation

Olson, Mallory

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Slow

Description

The extent to which a node is close to the other nodes both in the network and in space

Input :

Agent by Agent : DataType=real.

Agent by Location : DataType=real.

Output :

Node Level : Type=agent and DataType=real

Closeness Network Centralization

Go To Measures by Report

Informal Name

None

Citation

Freeman, L.C. (1979). Centrality in Social Networks I: Conceptual Clarification. Social Networks, 1, 215-239.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Slow

Description

Network centralization based on the closeness centrality of each node in a square network. This is defined only for connected, undirected networks.

Example : An organization has employees, computers, benchmarks, and products. How central these entities exist within the connections of the network, which is defined by the connections they have throughout the network, is loosely what is network centralization. This measure computes a value for the centrality of these entities.

Input :

Square matrix : DataType=real.

Output :

Graph Level : DataType=real.

$$\Re \in [0,1]$$

(if N connected)

Let **G=(V,E)** represent the square network, and let **n = |V|**

let

$$d_i = \text{Closeness Centrality of node i}$$

let

$$\overline{d} = \max\{d_i \mid 1 \le i \le n\}$$

Then

$$\text{Network Closeness Cent.} = \left(\sum_{1 \leq i \leq n} \overline{d} - d_i \right) / ((n-2)(n-1)/(2n-3))$$

Cognitive Demand

Go To Measures by Report

Informal Name

Emergent Leader

Citation

Carley, Kathleen M. & Yuqing Ren, 2001, "Tradeoffs Between Performance and Adaptability for C3I Architectures." In Proceedings of the 2001 Command and Control Research and Technology Symposium, Annapolis, Maryland, June, 2001.

Carley, 2002.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

Measures the total amount of cognitive effort expended by each agent to do its tasks, need to move, connecting others, and so on. Such individuals may never become the formal leader of a group. Emergent leaders are identified in terms of the amount of cognitive effort that is inferred to be expended based on the individual's position in the meta-network. Individuals who are strong emergent leaders are likely to be not just connected to many people, organizations, tasks, events, areas of expertise, and resources; but also, are engaged in complex tasks where they may not have all the needed resources or knowledge and so have to coordinate with

others, or have other reasons why they need to coordinate or share data or resources.

Input :

Agent by Agent : DataType=binary.

Output :

Node Level : Type=agent and DataType=real.

$$\Re \in [0,1]$$

The Cognitive Demand for an agent i is an average of terms, each of which measures an aspect of its cognitive demand. Each term is normalized to be in [0,1]. The number of terms depends on the input networks. The computation of each term for agent i is detailed below:

x1 = normailzed count of all agents with which agent i interacts.

$$x_1 = \frac{\sum[AA]_i}{|A| - 1}$$

x2 = normalized count of tasks to which i is assigned

$$x_2 = \frac{\sum[AT]_i}{|T|}$$

x3 = normalized sum of agents assigned to the same tasks as i

$$[ATA] = [AT] \cdot [AT]'$$

$$x_3 = \frac{\sum[ATA]_i - \sum_j[ATA]_{i,j}}{|T|(|A| - 1)}$$

x4 = normalized count of resources managed by i

$$x_4 = \frac{\sum[AR]_i}{|R|}$$

x5 = normalized count of total knowledge known by i

$$x_5 = \frac{\Sigma[AK]_i}{|K|}$$

x6 = normalized sum resources needed by i for all of its tasks

$$[ATR] = [AT] \cdot [RT]'$$

$$x_6 = \frac{\Sigma[ATR]_i}{|T||R|}$$

x7 = normalized sum of knowledge i needs for all its tasks

$$[ATK] = [AT] \cdot [KT]'$$

$$x_7 = \frac{\Sigma[ATK]_i}{|T||K|}$$

x8 = normalized sum of resource negotiation needs i has for its tasks

$$x_8 = \frac{HD([AR]_i, [ATR]_i)}{|R|}$$

x9 = normalized sum of knowledge negotiation needs

$$x_9 = \frac{HD([AK]_i, [ATK]_i)}{|K|}$$

x10 = normalized sum of agents that i depends on or that depend on i

$$w = colsum([AT])$$

$$s = (T+T') \bullet w$$

$$v = [AT] \bullet s$$

$$x_{10} = \frac{v_i}{|A||T|(|T|-1)}$$

Then Cognitive Demand for agent i is the average of the above terms.

Cognitive Distinctiveness

Informal Name

None

Citation

Carley, Kathleen M. 2002. "Summary of Key Network Measures for Characterizing Organizational Architectures." Unpublished Document: CMU 2002.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Slow

Description

Measures the degree to which each pair of agents has complementary knowledge, expressed as the percent of total knowledge.

Input :

Agent by Knowledge : DataType=binary.

Output :

Node Level : Type=agent and DataType=real.

Dyad Level : Agent by Agent and DataType=real.

$$\Re \in [0,1]$$

For each pair of agents (i,j) compute the number of knowledge bits they have exactly opposite. Then normalize this sum by the total

number of knowledge bits. In effect, this is the exclusive-OR of their knowledge vectors.

$$CD_{i,j} = \frac{\sum_{k=1}^{|K|}(AK_{i,k}*\sim AK_{j,k})+(\sim AK_{i,k}*AK_{j,k})}{|K|}$$

$$CD_{i,i} = 0$$

NOTE : The CD output matrix is symmetric.

🔳 Cognitive Expertise

Go To Measures by Report

Informal Name

None

Citation

Carley, Kathleen M. 2002. "Summary of Key Network Measures for Characterizing Organizational Architectures." Unpublished Document: CMU 2002

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Slow

Description

Measures the degree to which each pair of agents has complementary knowledge, expressed as a percentage of the knowledge of the first agent.

Example : If one person in an organization knows how do perform X but can't do Y. Whereas another individual can do Y but not X, such individuals would rank highly in this measure.

Input :

Agent by Knowledge : DataType=binary.

Output :

Node Level : Type=agent and DataType=real.

Dyad Level : agent by agent with DataType=real.

$$\Re \in [0,1]$$

For each pair of agents (i,j) compute the number of knowledge bits that j knows that i does not know. Then normalize this sum by the total number of knowledge bits that agent i does not know.

$$CE_{i,j} = \frac{\sum_{k=1}^{|K|} (\sim AK_{i,k} * AK_{j,k})}{(|K| - \sum_{k=1}^{|K|} AK_{i,k})}$$

$$CE_{i,i} = 0$$

NOTE : The CD output matrix is NOT-symmetric.

Cognitive Resemblance

Go To Measures by Report

Informal Name

None

Citation

Carley, Kathleen M. 2002. "Summary of Key Network Measures for Characterizing Organizational Architectures." Unpublished Document: CMU 2002.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Slow

Description

Measures the degree to which each pair of agents has the exact same knowledge. Each value is normalized by the row sum.

Example : In an given organizational network, let us say that Bob has the knowledge to perform X, Y, and Z. Whereas Sally also has the knowledge to perform X, Y and Z. This measure would rank this pair (i.e., dyad) high in Cognitive Resemblance.

Input :

Agent by Knowledge : DataType=binary.

Output :

Dyad Level : Agent by Knowledge and DataType=real.

$$\Re \in [0,1]$$

For each pair of agents (i,j) compute the number of knowledge bits they have in common - whether known or unknown. Then normalize this sum by the total number of knowledge bits.

$$CR_{i,j} = \frac{\sum_{k=1}^{|K|}(AK_{i,k} * AK_{j,k}) + (\sim AK_{i,k} * \sim AK_{j,k})}{|K|}$$

$$CR_{i,i} = 1$$

Cognitive Similarity

Go To Measures by Report

Informal Name

None

Citation

Carley, Kathleen M. 2002. "Summary of Key Network Measures for Characterizing Organizational Architectures." Unpublished Document: CMU 2002.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Slow

Description

Measures the degree to which each pair of agents have overlapping knowledge.

Example : With an organization where Bob has the knowledge of X, Y and Z and Sally has the knowledge of Q, R and Z. The Cognitive Similarity mesure would compute a value for Bob and Sally's overlapping knowledge of Z.

Input :

Agent by Knowledge : DataType=binary.

Output :

Node Level : Type=agent and DataType=real.

Dyad Level : Agent by Agent and DataType=real.

$$\Re \in [0,1]$$

For each pair of agents (i,j) compute the number of knowledge bits they have in common. Then normalize this sum by the total knowledge between them.

$$CS_{i,j} = \frac{\sum_{k=1}^{|K|}(AK_{i,k} * AK_{j,k})}{\sum_{k=1}^{|K|}(AK_{i,k} + AK_{j,k})}\Bigg|$$

$$CS_{i,i} = 1$$

NOTE : The CS output matrix is symmetric.

Column Breadth

Go To Measures by Report

Informal Name

None

Citation

Carley, Kathleen M., 2002

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Fast

Description

The fraction of entities with column nodes with degree greater than one.

Example : In a given organization, we have the following entities: employees (agents), computers (resources), benchmarks (tasks). This measure tells us on how many links greater than one exist within these entity types.

Input :

DataType=binary.

Output :

Graph Level : DataType=real.

$$\Re \in [0,1]$$

Let d = the Column degree vector of the network N

Then, the Column Breadth of N is:

$$\frac{\sum_{i=1}^{|d|}(d_i > 1)}{|d|}$$

NOTE : Row breadth is identical, except that d is defined using the row degree, rather than the column degree.

Column Count

Informal Name

None

Citation

Carley, Kathleen M., 2002

Minimum and Maximum Values

Minimum : 0

Maximum : Number of Columns

Computational Time

Fast

Description

The number of column nodes in a network.

Input :

DataType=binary.

Output :

Graph Level : DataType=integer.

Result: $Z \geq 0$

Column Degree Centrality

Go To Measures by Report

Informal Name

See **In Degree Centrality**

Column Degree Network Centralization

Go To Measures by Report

Informal Name

None

Citation

NetStat

Minimum and Maximum Values

Minimum :

Maximum :

Computational Time

Fast

Description

A centralization based on the degree of the column nodes of a network.

Input :

DataType=real

Output :

Node Level : Type=columns and DataType=real.

$$\Re \in [0,1]$$

Let N be a network with n column entities.

let

$$d_j = \text{degree of column node j, } 1 \le j \le n$$

let

$$\overline{d} = \max\{d_j | 1 \le j \le n\}$$

Then

$$\text{Column Degree Network Centralization} = \left(\sum_{1 \le j \le n} \overline{d} - d_j \right) / (n)$$

Column Rendundancy

Informal Name

None

Citation

Carley, Kathleen M. 2002. Summary of Key Network Measures for Characterizing Organizational Architectures. Unpublished Document: CMU 2002.

Minimum and Maximum Values

Minimum : 0

Maximum : $\Re \in [0, (m\text{-}1)*n]$

Computational Time

Fast

Description

The mean number of column node links in excess of one.

Input :

DataType=binary.

Output :

Graph Level : DataType=real.

$$\Re \in \left[0, (m-1)*n \right]$$

Let M be the matrix representation for a network N of dimension m x n.

let

$$d_j = \max\{0, \text{sum}(M(:,j)) - 1\}$$

for

$$1 \leq j \leq n$$

this is the number of column entries in excess of one for column j.

Then

$$\text{Column Redundancy} = \left(\sum_{j=1}^{n} d_j \right) / n$$

Communication

Informal Name

None

Citation

Carley, Kathleen M., 2003

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

Measures the communication need of agents to complete their assigned tasks.

Example : In a given organization **Bob** is assigned a task which is dependant upon his social links to other individuals in the network. The measure would compute a value for how dependent Bob is on the links to the other individuals to complete the task. A higher value would indicate Bob must possess a greater number of connections to complete any given

task he is assigned. Conversely, a low value would indicate relatively few linkages are necessary to complete certain tasks.

Input :

Agent by Agent : DataType=binary.

Agent by Task : DataType=binary.

Agent by Resource : DataType=binary.

Resource by Task : DataType=binary.

Task by Task : DataType=binary.

Output :

Node Level : Type=agent and DataType=real.

$$\Re \in [0,1]$$

Communication uses the concepts from Communication Congruence: **Handoff, Co-Assignment, and Negotiation.**

let H, C, and N be defined as in Communication Congruence.

let M(i,j) = [AA + (H+H') + C + (N+N')](i,j) > 0, and M(i,i) = 0

NOTE : The transpose of H and N is used to make the communication reciprocal.

let d = sum(M(i,:))

let d = d / (|A|-1), normalizing d to be in [0,1]

Then Communication for agent i is d.

Communication Congruence

Go To Measures by Report

Informal Name

None

Citation

Carley, Kathleen M., 2002. "Summary of Key Network Measures for Characterizing Organizational Architectures." Unpublished Document: CMU 2002

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

Measures to what extent the agents communicate when and only when it is needful to complete tasks. Perfect congruence requires a symmetric Agent x Agent network.

One of the following pairs of matrices must exist: **AK/KT, AR/RT**. If both exist, then the measure first concatenates them into **[AK AR], [KT RT]** and uses them.

Communication Congruence = 1 iif agents communicate when and only when it is needful to complete their tasks. There are three task related reasons when agents i and j need to communicate:

(a) **Handoff:** if i is assigned to a task s and j is assigned to a task t and s directly precedes task t.

(b) **Co-Assignment:** if i is assigned to a task s and j is also assigned to s.

(c) **Negotiation:** if i is assigned to a task s and j is not, and there is a resource r to which agents assigned to s have no access but j does.

The three cases are computed as follows:

(a) let $H = AT*TT*AT'$

(b) let $C = AT*AT'$

(c) let $N = AT*Z*AR'$, where $Z(t,r) = [AT'*AR - RT'](t,r) < 0$

let Q(i,j) = [(H+H') + C + (N+N')](i,j) > 0.

Communication Congruence requires reciprocal communication, explaining the transposes of H and N to make them symmetric.

let d = hamming distance between Q and AA, which measures the degree to which communication differs from that which is needed to do tasks.

The maximum value for d is d_max = |A|*(|A|-1)

Then Communication Congruence = 1 - (d /d_max), which is in [0,1].

Input :

Agent by Agent : DataType=binary.

Agent by Task : DataType=binary.

Agent by Resource : DataType=binary.

Resource by Task : DataType=binary.

Task by Task : DataType=binary.

Output :

Graph Level : DataType=real.

$$\Re \in [0,1]$$

Communicative Need

Go To Measures by Report

Informal Name

See Reciprocal Link Count for full description

Citation

Carley Kathleen M., 2002

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

Measures the percentage of reciprocal links in a network. This is also called Reciprocity.

Input :

Square matrix : DataType=binary

Output :

Graph Level : DataType=real.

Complexity

Go To Measures by Report

Informal Name

None

Citation

Wasserman and Faust, 1994

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Fast

Description

The density of the meta-network as a whole.

Input :

DataType=binary.

Output :

Graph Level : DataType=real.

Connectedness

Informal Name

None

Citation

Krackhardt, D. 1994. Graph Theoretical Dimensions of Informal Organizations. In Computational Organization Theory, edited by Carley, Kathleen M. and M.J. Prietula. Hillsdale, NJ: Lawrence Erlbaum Associates, 1994.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

Measures the degree to which a square network's underlying (symmetrized) network is connected. Individuals or organizations that connect groups are those that are connecting disconnected groups. The individual or organization may be connected to only one or a few members of each groups. This is a composite measure that is high when the agent is

potentially influential but is not in the know. This is calculated as the ration of betweenness centrality to total degree centrality and is calculated on agent by agent matrices.

Input :

Square matrix : DataType=binary.

Output :

Graph Level : DataType=real.

Degree Centrality Spatial

Informal Name

None

Citation

Olson, Mallory

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Slow

Description

The total distance from a node to its neighbors

Input :

Agent by Agent : DataType=real.

Agent by Location : DataType=real.

▦ Density

Go To Measures by Report

Informal Name

None

Citation

Wasserman, Stanley and Katherine Faust. Social Network Analysis: Methods and Applications. Cambridge: Cambridge University Press, 1994.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Fast

Description

The ratio of the number of links versus the maximum possible links for a network.

Density compares existing links to all possible links in the employee communication network. It reflects the social level of organizational cohesion. This measure must be interpreted in relation to the size of the group and the type of work performed.

Input :

Unimodel matrix : DataType=binary.

Output :

Graph Level : DataType=real.

$$\Re \in [0,1]$$

Let M be the adjacency matrix for the network of dimension m x n.

If the network is unimodal, then m=n and M has a zero diagonal, and therefore

Density = sum(M)/(m*(m-1)).

For rectangular networks

Density = sum(M)/(m*n).

Density Classic SNA

Go To Measures by Report

Informal Name

None

Citation

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Fast

Description

The ratio of the number of links versus the maximum possible links for a square network. Self-loops are ignored.

Input :

Square matrix : dataType=binary.

Output :

Graph Level : DataType=real.

$$\Re \in [0,1]$$

Let M be the adjacency matrix for the network of dimension m x n.

If the network is unimodal, then m=n and M has a zero diagonal, and therefore

Density = sum(M)/(m*(m-1)).

For rectangular networks, Density = sum(M)/(m*n).

Diameter

Go To Measures by Report

Informal Name

None

Citation

Wasserman, Stanley and Katherine Faust. Social Network Analysis: Methods and Applications. Cambridge: Cambridge University Press, 1994.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

The maximum shortest path length between any two nodes in a unimodal network G=(V,E). If there exist i,j in V such that j is not reachable from i, then |V| is returned.

Input :

Square matrix : DataType=binary.

Output :

Graph Level : DataType=integer.

$$Z \in [0, |V|]$$

The diameter of G=(V,E) is defined as:

$$\max\{d_G(i, j) \mid i, j \in V\}$$

That is, the maximum shortest directed path between any two vertices in G. If there exists i and j such that j is not reachable from i, then |V| is returned.

Diffusion

Go To Measures by Report

Informal Name

None

Citation

Carley, Kathleen M., 2002

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Slow

Description

Computes the degree to which something could be easily diffused (spread) throughout the network. This is based on the distance between nodes. A

large diffusion value means that nodes are close to each other, and a smaller diffusion value means that nodes are farther apart.

Input :

Square matrix : DataType=binary.

Output :

Graph Level : DataType=real.

Let N = number of nodes.

Let K = number of pairs of nodes that are unreachable (note that G is not required to be symmetric.

For each pair of unreachable nodes we record a distance of N, which is beyond the maximum distance for a binary network.

Let TotalDistance = $\sum_{i=j} d_G(i,j) + K*N$

Let AverageDistance = TotalDistance / (N*(N-1)), which is the average distance between nodes

Then Diffusion = N/(N-1) * [1 - AverageDistance/N], which is the average distance from 1 and normalized so that cliques get a score of 1

Distinctive Correlation

Go To Measures by Report

Informal Name

None

Citation

Carley, Kathleen M., 2002

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Description

Measures the degree to which each pair of rows has complementary data, expressed as the percent of total data.

Input :

DataType=binary.

Output :

Node Level : Type=row and DataType=real.

Dyad Level : Row by Row and DataType=real.

Edge Betweenness Centrality

Go To Measures by Report

Informal Name

None

Citation

Freeman, 1979

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Slow

Description

The Edge Betweenness Centrality of an edge e in a network is defined as: across all node pairs that have a shortest path containing e, the percentage that pass through e.

Input :

Agent by Resource : DataType=binary.

Output :

Graph Level : DataType=real.

$$\Re \in [0, (|A| - 1) * |R|]$$

This is the Column Redundancy of matrix AR.

Efficiency

Go To Measures by Report

Informal Name

None

Citation

Krackhardt, D. 1994. Graph Theoretical Dimensions of Informal Organizations. In Computational Organization Theory, edited by Carley, Kathleen M. and M.J. Prietula. Hillsdale, NJ: Lawrence Erlbaum Associates, 1994.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Fast

Description

The degree to which each component in a network contains the minimum links possible to keep it connected.

Example : In an organization, let us say Bob has a connection to the computer, Sales, Event, and Goal. This measure computes a value for the minimum number of links entities such as Bob must have for the network to maintain its network status. Therefore, this is a network level measure. In other words, if Bob had one less link, then then the entities would cease to constitute the original network.

Input :

Square : DataType=binary and Directed=No.

Output :

Graph Level : DataType=real.

$$\Re \in [0,1]$$

Let **G=(V,E)** be the graph representation of a square, undirected network.

let

$$n = \text{number of components in G}$$

let

$$c_i = \text{number of nodes in component i}$$

let

$$\text{penalty} = |E| - |V| + C$$

let

$$\text{maxPenalty} = C - |V| + \sum_i c_i (c_i - 1) / 2$$

Then **Efficiency = 1 - penalty/maxPenalty**

Eigenvector Centrality

Go To Measures by Report

Informal Name

None

Citation

Bonacich P, 1972

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

Calculates the principal eigenvector of the network. A node is central to the extent that its neighbors are central. Leaders of strong cliques are individuals or organizations who are connected to others that are themselves highly connected to each other. In other words, if you have a clique then the individual most connected to others in the clique and other cliques, is the leader of the clique. Individuals or organizations who are connected to many otherwise isolated individuals or organizations will have a much lower score in this measure then those that are connected to groups that have many connections themselves. This would be true even if such a **WHO** might have a high amount of actual connections **(i.e. high degree centrality)**.

It calculates the eigenvector of the largest positive eigenvalue of the adjacency matrix representation of a square network. A Jacobi method is used to compute the eigenvalues and vectors.

Hub-Centrality and Authority-Centrality measures are generalizations of the Eigenvector Centrality measure and take into account directed links.

Example : It follows that a person well-connected to well-connected people can spread information much more quickly than one who only has connections to lesser important people in a network. People with higher scores of Eigenvector Centrality could be critical when rapid communication is needed.

Furthermore, individuals or organizations that are connected to man other otherwise isolated groups will have a much lower score in this measure

than those that are connected to groups that have many connections themselves.

Input :

Agent by Agent Node Level : DataType=real and Directed=No.

Output :

Node Level and DataType=real.

$$\Re \in [0,1]$$

Calculates the eigenvector of the largest positive eigenvalue of the adjacency matrix representation of a square network. A Jacobi method is used to compute the eigenvalues and vectors.

Eigenvector Centrality Spatial

Go To Measures by Report

Informal Name

None

Citation

Olson, Mallory

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Slow

Description

The sum of the eigenvector centralities of all nodes located at a place. Intuitively, the likelihood of arriving at a node at a particular location in a random walk.

Input :

Agent by Agent matrix : DataType=real.

Agent by Location matrix : DataType=real.

Output :

Node Level : Type=location and DataType=real

Exclusivity

Go To Measures by Report

Informal Name

None

Citation

Ashworth, 2002

Minimum and Maximum Values

Minimum :

Maximum :

Computational Time

Fast

Description

Detects entities that have ties that comparatively few other entities have. Individuals or organizations that are high in complete exclusivity for knowledge are those that have expertise or are connected to types of knowledge that no one else has. The scientific name of this measure is complete exclusivity and it is calculated on agent by knowledge matrices.

- Individuals or organizations that are high in complete exclusivity for location are those that have frequented locations that no one else has.

- Individuals or organizations that are high in complete exclusivity for resources are those that have resources or are connected to types of resources that few others have.

- Individuals or organizations that are high in complete exclusivity for tasks are those that are engaged in or assigned to tasks or activities for which no one else is engaged in or assigned to.

Input :

DataType=binary.

Output :

Node Level : Type=row and DataType=real.

Exclusivity Complete

Go To Measures by Report

Informal Name

None

Citation

Ashworth, M. and Kathleen M. Carley, 2003 "Critical Human Capital" Working Paper, CASOS, Carnegie Mellon, Pittsburgh PA.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Fast

Description

Detects entities that have ties that no other entity has. Individuals or organizations that are high in complete exclusivity for events are those that attend or are supposed to attend an event that no one else does. The

scientific name of this measure is complete exclusivity and it is calculated on agent by event matrices.

Example : Individuals or organizations that are high in complete exclusivity for events are those that attend or are supposed to attend an event that no one else does.

Input :

DataType=binary.

Output :

Node Level : Type=row and DataType=real.

$$\Re \in [0,1]$$

This is a variation of the Exclusivity measure that weights the edges of an entity as follows: if no other entity has a connection to the entity, then its weight is one, otherwise it is zero.

Thus, the Complete Exclusivity value for node i is:

$$\sum_{j=1}^{|K|} AK(i,j) * [\text{sum}(AK(:,j) == 0]$$

The value is then normalized by dividing by |K|.

Expertise Correlation

Go To Measures by Report

Informal Name

None

Citation

Carley, Kathleen, M., 2002

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Slow

Description

Measures the degree to which each pair of rows has complementary data, expressed as a percentage of the data of the first row.

Input :

DataType=binary.

Output :

Node Level : Type=row and DataType=real.

Dyad Level : Row by Row and DataType=real.

Fragmentation

Go To Measures by Report

Informal Name

None

Citation

Borgatti, S.P. 2003. The Key Player Problem. Dynamic Social Network Modeling and Analysis: Workshop Summary and Papers, R.

Breiger, K. Carley, & P. Pattison (Eds.) Committee on Human Factors, National Research Council, 241-252.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

The proportion of nodes in a network that are disconnected.

Example : In a given organization 25 out of 400 sales agents are disconnected from the resources and knowledge that exists at the company headquarters because they travel more frequently and are thus disconnected from the rest of the network. The attribute of this disconnection expressed as a property of the entire organization is its computed disconnected measure.

Input :

Square : DataType=binary and Directed=No.

Output :

Graph Level : DataType=real.

$$\Re \in [0,1]$$

Consider a square, symmetric network G=(V,E).

let

$$n = |V|$$

let

s_k be the number of nodes in the kth component of G, $1 \leq k \leq n$

Then,

$$\text{Fragmentation} = 1 - \frac{\sum_k s_k(s_k - 1)}{n(n-1)}$$

Geodesic Distance

Informal Name

None

Citation

Wasserman and Faust, 1994

Minimum and Maximum Values

Minimum : 0

Maximum : Number of columns - 1

Computational Time

Medium

Description

Computes the length of the shortest path between all node pairs, where length is the number of links, ignoring link weights. If no path exists between two nodes, then a distance of zero is given. The distance from a node to itself is also zero.

Input :

Square matrix : DataType=binary.

Output :

Dyad Level : DataType=integer.

Global Efficiency

Informal Name

None

Citation

Latria and Marchiori, 2001

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Slow

Description

Global Efficiency is the normalized sum of the inverse geodesic distances between all entity pairs.

Example : In an organization certain resources are closer to each other in terms of links. If both resources are located closer to each other than it would follow an organization where such resources tend to be more closely distributed across a network would be a more efficient network.

Input :

Square matrix : DataType=binary.

Output :

Graph Level : DataType=real.

$$\Re \in [0,1]$$

Global Efficiency is the normalized sum of the inverse geodesic distances between all entity pairs. Let G be a square network on n entities.

Then,

$$\text{Global Efficiency} = \frac{1}{n(n-1)} \sum_{i \neq j} \frac{1}{d_G(i,j)}$$

 Hierarchy

Informal Name

None

Citation

Krackhardt, D. 1994. Graph Theoretical Dimensions of Informal Organizations. In Computational Organization Theory, edited by Carley, Kathleen M. and M.J. Prietula. Hillsdale, NJ: Lawrence Erlbaum Associates, 1994.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Slow

Description

The degree to which a unimodal network exhibits a pure hierarchical structure.

Example : Let us say in an organization Sue is the president. Under Sue, we have 4 vice presidents, whom all in turn have 5 or 10 direct reports, who all have 2 or 3 managers working for them. To such a degree that this architecture is present in the organzatni is its computed Krackhardt Hierarchy value.

Input :

Square matrix : DataType=binary.

Output :

Graph Level : DataType=real.

$$\Re \in [0,1]$$

Let N be a square network. The Hierarchy of N is the Reciprocity of the Reachability Network for N.

Hub Centrality

Informal Name

None

Citation

Kleinberg J.M, 1998

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

A node is hub-central to the extent that its out-links are to nodes that have many in-links. Individuals or organizations that act as hubs are sending information to a wide range of others each of whom has many others reporting to them. Technically, an agent is hub-central if its out-links are to agents that have many other agents sending links to them. The scientific name of this measure is hub centrality and it is calculated on agent by agent matrices.

Input :

Square matrix : DataType=real.

Output :

Node Level : DataType=real.

Let G be the adjacency matrix.

Hub Centrality = eigenvectorCentrality(G*G')

Authority Centrality = eigenvectorCentrality(G'*G)

 In Degree Centrality

Informal Name

None

Citation

Wasserman, Stanley and Katherine Faust. "Social Network Analysis: Methods and Applications." Cambridge: Cambridge University Press, 1994.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Fast

Description

The In Degree Centrality of a node is its normalized in-degree. For any node, e.g. an individual or a resource, the in-links are the connections that the node of interest receives from other nodes. For example, imagine an agent by knowledge matrix then the number of in-links a piece of knowledge has is the number of agents that are connected to. The scientific name of this measure is in-degree and it is calculated on the agent by agent matrices.

For any node, e.g. an individual or a resource, the in-links are the connections that the node of interest receives from other nodes.

Example : An agent by knowledge matrix with a number of in-links. A piece of knowledge has is the number of agents that are connected to.

Input :

Agent by Agent : DataType=real.

Output :

Node Level : Type=column and DataType=real.

$$\Re \in [0,1]$$

Consider the matrix representation X of a network with m rows and n columns. Each column j is assigned an in-degree centrality based on the sum of entries in its column.

The In-Degree Centrality for column j

$$\text{column } j = \frac{1}{m} \sum_{i=1}^{m} X(i, j)$$

NOTE : This measure is also called Column Degree Centrality.

In Degree Network Centralization

Go To Measures by Report

Informal Name

None

Citation

NetStat

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Fast

Description

A centralization of a square network based on the In-Degree Centrality of each node.

485

Example : In a given organization, all employees have many incoming links to other employees. This is their **In degree** connections. This measure computes a value for the entire organization as a whole as to how central these connections are within the organization.

Input :

Square matrix : DataType=real.

Output :

Graph Level : DataType=real.

$$\Re \in [0,1]$$

Let **N be a unimodal network with n entities**.

let

$$d_i = \text{In Degree Centrality of node i}$$

let

$$\overline{d} = \max\{d_i \mid 1 \le i \le n\}$$

Then

$$\text{In Degree Network Centralization} = \left(\sum_{1 \le i \le n} \overline{d} - d_i \right) / D$$

where D = (n-2) if N is undirected, and (n-1) otherwise.

Information Centrality

Go To Measures by Report

Informal Name

None

Citation

Wasserman, Stanley and Katherine Faust. "Social Network Analysis: Methods and Applications." Cambridge: Cambridge University Press, 1994.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Slow, with High Space requirements

Description

Information centrality is a network metric that accounts for indirect as well as shortest (geodesic) paths among entities. Information centrality is similar to betweenness, except that betweenness considers only shortest paths **geodesics**, whereas information centrality also considers more circuitous paths weighted by the inverse of the path length **(the number of links along the path)**.

Calculates the Stephenson and Zelen information centrality measure for each entity.

Example : Entities high in information centrality are more likely to get more information and to get information faster.

Input :

Square matrix : DataType=real and Directed=No.

Output :

Node Level : DataType=real.

$$\Re \in [0,1]$$

Calculates the measure described on pg 195-6 of Wasserman and Faust. Isolates **(entities with no edges)** are first removed from the network, and the measure computed on the resulting sub-graph. The isolates are given centrality value 0.

Interdependence

Informal Name

None

Citation

Carley, Kathleen M., 2002. Summary of Key Network Measures for Characterizing Organizational Architectures. Unpublished Document: CMU 2002

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Fast

Description

The percentage of links in a unimodal network that are Pooled or Reciprocal.

Example : A reciprocal link is one that is bidirectional. Let us say Bob has a link to Sue and Sue has a link back to Bob This is a reciprocal link. However, if Bob had a lot of links to Sue and others, which were not reciprocated, that is bidirectional, than Bob would be said to rank higher in radial links, which linked in one direction. The extent that a network exhibits these relationships is computed as the Interdependence and Radial value for the network.

Input :

Square matrix : DataType=binary.

Output :

Graph Level : DataType=real.

$$\Re \in [0,1]$$

Let **G=(V,E)** be the graph representation of a square network.

Let **a = Pooled Edge Count** and **b = Reciprocal Link Count** of the network.

Then **Interdependence = (a+b)/|E|**

Interlockers

Informal Name

None

Citation

Carley, Kathleen M., 2002

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Fast

Description

Interlocker are nodes that have a triad count (the number of triads each node is in) that is greater than the mean plus one standard deviation.

Example : A typical corporate board of directors is a good example of an interlocking architecture. Whereas a predominant amount of board members - perhaps close to 70-percent - might serve on one board only, it is often common that the remaining 30-percent may serve on 3 or more other boards. These directors that serve on the other boards constitute the interlocking agents that connect all these typical boards together. The Interlockers and radial measures compute a value for such relationships as they exist in the network at the entity, that is agent (e.g., "board member"), level.

Input :

Square matrix : DataType=binary.

Output :

Node Level : DataType=binary.

Let **N=(V,E)** be a square network.

Let

t_i = Triad Count for node i, $1 \leq i \leq |V|$

Let

u = the mean of $\{ t_i \}$

Let

d = the variance of $\{ t_i \}$

Then,

if $t_k \geq (u + d)$, then agent k is an *interlocker*.

If $t_k \leq (u - d)$, then agent k is a *radial*.

Inverse Closeness Centrality

Go To Measures by Report

Informal Name

None

Citation

Wasserman, Stanley and Katherine Faust. Social Network Analysis: Methods and Applications. Cambridge: Cambridge University Press, 1994.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Slow

Description

The average closeness of a node to the other nodes in a network. Inverse Closeness is the sum of the inverse distances between a node and all other nodes. Nodes high in closeness centrality are more likely to communicate faster and operate more efficiently.

Input :

Square matrix : DataType=real.

Output :

Node Level : DataType=real.

$$\Re \in [0,1]$$

Let **G=(V,E)** be the graph representation of the square network.

For $v \in V$

let

$$\text{dist} = \sum_{i \in V} \frac{1}{d_G(v,i)} \text{, where } \frac{1}{d_G(i,i)} = 0 \text{ and } \frac{1}{d_G(v,i)} = 0$$

if i is not reachable from v.

Then, **Inverse Closeness Centrality of entity v = dist/(|V|-1)**.

 # Isolate Count

Informal Name

None

Citation

Carley, Kathleen, M., 2002

Minimum and Maximum Values

Minimum : 0

Maximum : Number of nodes in the network

Computational Time

Fast

Description

The number of isolate nodes in a unimodel network.

Input :

Square matrix : DataType=binary.

Output :

Graph Level : DataType=integer.

$$Z \in [0, |V|]$$

Let N=(V,E) be a square network.

An isolate is a node v in V such that no edges are incident upon it.
The number of such nodes is returned.

Knowledge Actual Workload

Go To Measures by Report

Informal Name

None

Citation

Carley, Kathleen M., 2002. "Summary of Key Network Measures for Characterizing Organizational Architectures." Unpublished Document: CMU 2002

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

The knowledge an agent uses to perform the tasks to which it is assigned.

Input :

Agent by Knowledge : DataType=binary

Knowledge by Task : DataType=binary

Agent by Task : DataType=binary

Output :

Node Level : Type=agent and DataType=real.

Knowledge Based Access Index

Go To Measures by Report

Informal Name

None

Citation

Ashworth, M. and Carley, Kathleen M., 2003, "Critical Human Capital", Working Paper, CASOS, Carnegie Mellon, Pittsburgh PA.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Fast

Description

Boolean value which is true if an agent is the only agent who knows a piece of knowledge and who is known by exactly one other agent. The one agent known also has its KAI set to one.

An node level measure that computes the average shortest path length between entities, excluding infinite distances. This is known as the Knowledge Access Index (KAI) value.

Example : Let us imagine a social network where 50 employees have many connections to each other. The average number of connections per employee is the Access Index.

Input :

Agent by Agent : DataType=binary

Agent by Knowledge : DataType=binary

Output :

NodeLevel : Type=agent and DataType=binary.

The Knowledge Access Index (KAI) for agent i is defined as follows:

let

$$ S_i = \{s \mid AK(i, s) \wedge (\text{sum}(AK(:, s)) = 1) \wedge (\text{sum}(A(i, :)) = 1)\} $$

Then

$$KAI_i = \left(\left(S_i \neq \varnothing \right) \vee \left(\exists j \,|\, S_j \neq \varnothing \wedge A(j,i) = 1 \right) \right)$$

Knowledge Based Task Completion

Informal Name

None

Citation

Carley, Kathleen M., 2002. Summary of Key Network Measures for Characterizing Organizational Architectures. Unpublished Document: CMU 2002

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

The percentage of tasks that can be completed by the agents assigned to them, based solely on whether the agents have the requisite knowledge to do the tasks.

Example : Bob is assigned 10 Tasks to complete which require a certain amount of knowledge to finish. However, on average only 10-percent of this knowledge will be available to Bob for him to complete the tasks given. This measure would compute a value for the likelyhood that Bob would be able to complete any given task.

NOTE : Task Completion Overall is a combination of the Task Completion Knowledge and the Task Completion Resource.

Input :

Agent by Knowledge : DataType=binary.

Agent by Task : DataType=binary.

Knowledge by Task : DataType=binary.

Output :

Graph Level : DataType=real

$$\Re \in [0,1]$$

Find the tasks that cannot be completed because the agents assigned to the tasks lack necessary knowledge.

let

$$Need = [(AT'*AK) - KT']$$

let

$$S = \{\ i\ |\ 1 \leq i \leq |T|,\ \exists\ j : Need(i,j) < 0\ \}$$

Knowledge Based Task Completion is the percentage of tasks that could be completed = (|T|-|S|) / |T|

Resource Based Task Completion is analogous, replacing AK with AR and KT with RT.

Knowledge Diversity

Informal Name

None

Citation

Borgatti, S.P. 2003. The Key Player Problem. Dynamic Social Network Modeling and Analysis: Workshop Summary and Papers, R.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Fast

Description

The distribution of difference in idea sharing. This is the Herfindahl-Hirshman index applied to column sums of AK.

Example : Characterizes the network in terms of how the items in a knowledge network **(who knows what)** are non-uniformly distributed across employees

   Input :

   Agent by Knowledge : DataType=binary.

Output :

   Graph Level : DataType=real

   $$\Re \in [0,1]$$

   This is the Herfindahl-Hirshman index (economics: sum of the squares of each firm's market share) applied to the normalized column sums of AK. This measures the degree to which knowledge is equally known.

   let

   $$w_k = \sum_{i=1}^{|A|} AK(i,k) \text{ , for } 1 \leq k \leq |K|$$

   let

   $$W = \sum_{k=1}^{|K|} w_k$$

   Then

$$\text{Knowledge Diversity} = 1 - \sum_{k=1}^{|K|} \left(w_k / W \right)^2$$

Knowledge Exclusivity

Go To Measures by Report

Informal Name

None

Citation

Ashworth, 2002

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Fast

Description

Detects agents who have singular knowledge.

Example : A key piece of knowledge that supports an entire organization is only accessible through one person.

Input :

Agent by Knowledge : DataType=binary.

Output :

Node Level : DataType=real and Type=agent.

$\Re \in [0,1]$

The **Knowledge Exclusivity Index (KEI)** for agent i is defined as follows:

$$\sum_{j=1}^{|K|} AK(i,j) * \exp(1 - \text{sum}(AK(:,j)))$$

The **Knowledge Exclusivity Index** (KEI) for agent i is defined exactly as for Knowledge Based Exclusivity.

Knowledge Load

Go To Measures by Report

Informal Name

None

Citation

Carley, Kathleen M., 2002. "Summary of Key Network Measures for Characterizing Organizational Architectures." Unpublished Document: CMU 2002.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Fast

Description

Average number of knowledge per agent.

Input :

Agent by Knowledge : DataType=binary

Output :

Graph Level : DataType=real.

$$\Re \in \left[0, \left|K\right|\right]$$

Knowledge Load = sum(AK)/ (|A|)

Resource Load is defined analogously, replacing AK with AR.

Knowledge Negotiation

Informal Name

None

Citation

Carley, Kathleen M., 2002. "Summary of Key Network Measures for Characterizing Organizational Architectures." Unpublished Document: CMU 2002

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

The extent to which agents need to negotiate with each other because they lack the knowledge to complete their assigned tasks.

Example : In a given organization, Bob does not have knowledge needed to complete a certain task but Sue does have it. Bob must negotiate with Sue to gain that knowledge to complete the task. The extent to which others in the organization must negotiate for knowledge that others have is the organization's knowledge negotiation value.

Input :

Agent by Task : DataType=binary.

Agent by Knowledge : DataType=binary.

Knowledge by Task : DataType=binary.

Output :

Graph Level : DataType=real.

$$\Re \in [0,1]$$

Compute the percentage of tasks that lack at least one knowledge *(or Resource)*:

let

$$Need = (AT'*AK) - KT'$$

for Resource replace AK with AR and KT with RT

let

$$S = \{ \, i \mid 1 \leq i \leq |T| \,, \; \exists \; j : Need(i,j) < 0 \, \}$$

Then **Knowledge (or Resource) Negotiation = |S| / |T|**

Knowledge Omega

Go To Measures by Report

Informal Name

None

Citation

Carley, K, Dekker, D., Krackhardt, D (2000). How Do Social Networks Affect Organizational Knowledge Utilitization.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

The degree to which agents reuse knowledge while doing their tasks.

Example : Sue has X amount of tasks within her company. How many of those tasks that require a knowledge that can be used for one or more of her other tasks is omega knowledge. This measure computes a value for such relations to knowledge sets at the network level.

Input :

Agent by Task : DataType=binary.

Knowledge by Task : DataType=binary.

Task by Task : DataType=binary.

Output :

Graph Level : DataType=real.

$$\Re \in [0,1]$$

Let **TAT = TA*TA'**

Let **N = ((T'@TAT)*KT')@KT'**

for resource replace KT with RT

Then **Knowledge Based Omega = sum(N)/sum(KT)**

for resource replace KT with RT

Knowledge Potential Workload

Informal Name

None

Citation

Carley, Kathleen M., 2002. Summary of Key Network Measures for Characterizing Organizational Architectures. Unpublished Document: CMU 2002.

Minimum and Maximum Values

Minimum :

Maximum :

Computational Time

Medium

Description

Maximum knowledge an agent could use to do tasks if it were assigned to all tasks.

Example : In a given organization, let us say Bob is assigned all task that exist in the organization. This measure would compute a value expressing Bob's potential to carry out all the tasks based on his connections to the resources and knowledge needed for the tasks.

Input :

Agent by Knowledge : DataType=binary.

Knowledge by Task : DataType=binary.

Output :

Node Level : Type=agent and DataType=real.

$$\Re \in [0,1]$$

Potential Knowledge Workload for agent i = sum((AK*KT)(i,:))/sum(KT)

for Resource replace AK with AR and KT with RT

Knowledge Rendundancy

Informal Name

None

Citation

Carley, Kathleen M., 2002. "Summary of Key Network Measures for Characterizing Organizational Architectures." Unpublished Document: CMU 2002.

Minimum and Maximum Values

Minimum : 0

Maximum : $\Re \in [0,(|A|-1)*|K|]$

Computational Time

Fast

Description

Average number of redundant agents per knowledge. An agent is redundant if there is already an agent that has the knowledge.

Input :

Agent by Knowledge : DataType=binary.

Output :

Graph Level : DataType=real.

$$\Re \in [0,(|A|-1)*|K|]$$

This is the Column Redundancy of matrix AK.

Knowledge Under Supply

Go To Measures by Report

Informal Name

None

Citation

Carley, Kathleen M., 2002. Summary of Key Network Measures for Characterizing Organizational Architectures. Unpublished Document: CMU 2002

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

The extent to which the knowledge needed to do tasks are unavailable in the entire organization.

Input :

Agent by Task : DataType=binary.

Agent by Knowledge : DataType=binary.

Knowledge by Task : DataType=binary.

Output :

Graph Level : DataType=real

$$\Re \in [0,1]$$

This is the number of required knowledge not available to tasks based on the current agent task assignments normalized by the number of tasks.

Knowledge Under Supply = sum(dich(KT - AK'*KT)) / |T|

Under Resource Supply is analogous, replacing AK with AR, and KT with RT.

Lateral Link Count

Go To Measures by Report

Informal Name

None

Citation

Carley, Kathleen M., 2002. Summary of Key Network Measures for Characterizing Organizational Architectures. Unpublished Document: CMU 2002.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Fast

Description

The percentage of lateral links in a unimodal network. Fixing a root node x, a lateral link (i,j) is one in which the distance from x to i is the same as the distance from x to j.

Example : Fixing a root entity x, a lateral edge (i,j) is one in which the distance from x to i is the same as the distance from x to j.

Input :

Square matrix : DataType=binary.

Output :

Graph Level : DataType=real

$$\Re \in [0,1]$$

Let **G=(V,E)** be the graph representation of a network. And fix an node **x∈V** to be the root node.

Let

$$\text{Let } S = \{(i,j) \in E \mid d_G(x,i) = d_G(x,j)\}$$

Then **Lateral Edge Count = |S| / |E|**

Link Count

Go To Measures by Report

Informal Name

None

Citation

Carley, Kathleen M., 2002

Minimum and Maximum Values

Minimum : 0

Maximum : Number of links in the network

Computational Time

Fast

Description

The number of links in the network.

Output :

Graph Level : DataType=integer.

Local Efficiency

Go To Measures by Report

Informal Name

None

Citation

Latora and Marchiori, 2001 (Full reference forthcoming)

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Slow

Description

Normalized sum of the Global Efficiency of each node's ego network (here, the ego network for entity i does not include entity i).

Example : In an organization, Bob has a connection to the computers, Sales Data, Events, Goals. This measure computes a value for the minimum number of links to entities such as Bob must have for the network to maintain its network status. Therefore, this is a network level measure. If Bob had one less link, then then the entities would cease to constitute the original network.

Input :

Square matrix : DataType=binary.

Output :

Graph Type : DataType=real.

$$\Re \in [0,1]$$

Local Efficiency is the normalized sum of the inverse closeness of the ego networks (here, the ego network for entity i does not include entity i).

Let **G = (V,E)** be any square, symmetric network with **|V| = N entities**.

let

$$V_i = \text{set of nodes adjacent to } i = \{ j \mid (i,j) \in E \}$$

let

$$E_i = \text{edges between nodes adjacent}$$

to

$$i = \{ (j,k) \mid (j,k) \in E; \, i,j \in V_i \}$$

define

$$G_i = (V_i, E_i)$$

Then,

$$\text{Local Efficiency} = \frac{1}{N} \sum_{i=1}^{N} \text{GlobalEfficiency}(G_i)$$

Location Relevance

Go To Measures by Report

Informal Name

None

Citation

Olson, Malloy

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Slow

Description

The extent to which location influences a particular persons connections.

Input :

Agent by Agent : DataType=real.

Agent by Location : DataType=real.

Output :

Node Level : Type=location and DataType=real.

Minimum Speed

Go To Measures by Report

Informal Name

None

Citation

Carley, Kathleen M., 2002. Summary of Key Network Measures for Characterizing Organizational Architectures. Unpublished Document: CMU 2002.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

The maximum shortest path length between node pairs (i,j) where there is a path in the network from i to j. If there are no such pairs, then Minimum Speed is zero.

Example : In an organization, we have Bob and Sue with numerous paths between them. The shortest path between this entity pair would be Bob and Sues (Bob/Sue) minimum speed. One might also say that if there were numerous links between Bob and Sue and that the average linkage was in the neighborhood of 5 paths (e.g., links) when every person in the organization is taken as a whole, the shortest linkage between them would be the minimum speed at which Bob and Sue might communicative. If Bob and Sue had direct links, minimum speed would be 0.

Input :

Square matrix : DataType=real.

Output :

Graph Level : DataType=real.

$$\Re \in [0,1]$$

Minimum Speed = 1 / (Levels for the Network)

Network Levels

Informal Name

None

Citation

Carley, Kathleen M., 2002

Minimum and Maximum Values

Minimum : 0

Maximum : Z∈[0,|V|-1]

Computational Time

Medium

Description

The Network Level of a square network is the maximum Node Level of its nodes.

Input :

Square matrix : DataType=real.

Output :

Graph Level : DataType=integer.

$$Z \in [0, |V| - 1]$$

Let **G=(V,E)** be the graph representation of a square network and fix an entity v.

Then

$$\text{the Levels of } G = \max \{ d_G(i, j) \mid i,j \in V; j \text{ reachable from } i \text{ in } G \}$$

Node Count

Go To Measures by Report

Informal Name

None

Citation

Carley, Kathleen M., 2002

Minimum and Maximum Values

>   **Minimum :** 0

>   **Maximum :** Number of nodes in the network

Computational Time

Fast

Description

The number of nodes in a unimodal network.

>   **Input :**

>>   Square matrix : DataType=binary.

>   **Output :**

>>   Graph Level : DataType=integer.

Node Levels

Informal Name

None

Citation

Carley, Kathleen M., 2002

Minimum and Maximum Values

Minimum : 0

Maximum : Number of nodes in the network - 1

Computational Time

Medium

Description

The Node Level for a node v in a square network is the longest shortest path from v to every node v can reach. If v cannot reach any node, then its level is 0.

The Network Level of a square network is the maximum Node Level of its nodes.

Input :

Square matrix : DataType=real.

Output :

Graph Level : DataType=integer.

$$Z \in [0, |V| - 1]$$

Let **G=(V,E)** be the graph representation of a square network and fix an node v.

Node Level : DataType=integer

$$\text{for } v = \max \{ d_G(v, j) \mid j \in V; j \text{ reachable from } v \text{ in } G$$

If v cannot reach any nodes, then its level is 0.

Organization Agent Knowledge Needs Congruence

Go To Measures by Report

Informal Name

None

Citation

Lee, 2004

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

Across all agents, the knowledge that agents lack to do their assigned tasks expressed as a percentage of the total knowledge needed by all agents.

Input :

Agent by Knowledge matrix : DataType=binary.

Agent by Task matrix : DataType=binary.

Knowledge by Task matrix : DataType=binary.

Output :

Graph Level : DataType=real.

$$\Re \in [0,1]$$

As in Agent Needs, let N = AT*KT'

Then

$$\text{Organization Agent Needs} = \frac{\sum_{i=1}^{|A|} \sum_{j=1}^{|K|} N_{i,j} * (\sim AK_{i,j})}{\text{sum}(N)}$$

Organization Agent Knowledge Waste Congruence

Go To Measures by Report

Informal Name

None

Citation

Lee, 2004

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

Across all agents, the knowledge that agents have that are not required to do their assigned tasks.

Input :

Agent by Knowledge matrix with DataType=binary.

Agent by Task matrix with DataType=binary.

Knowledge by Task matrix with DataType=binary.

Output : Graph Level with DataType=real.

$$\Re \in [0,1]$$

As in Agent Waste, let N = AT*KT'

Then

$$\text{Organization Agent Needs} = \frac{\sum_{i=1}^{|A|} \sum_{j=1}^{|K|} AK_{i,j} * (N_{i,j} = 0)}{\text{sum}(AK)}$$

Organization Agent Resource Needs Congruence

Informal Name

None

Citation

Lee, 2004

Minimum and Maximum Values

   Minimum : 0

   Maximum : 1

Computational Time

Medium

Description

Across all agents, the resource that agents lack to do their assigned tasks expressed as a percentage of the total resources needed by all agents.

   Input :

      Agent by Resource : DataType=binary.

      Agent by Task : DataType=binary.

      Resource by Task : DataType=binary.

   Output :

Graph Level : DataType=real.

$$\Re \in [0,1]$$

As in Agent Needs, let N = AT*RT'

Then

$$\text{Organization Agent Needs} = \frac{\displaystyle\sum_{i=1}^{|A|} \sum_{j=1}^{|R|} N_{i,j} * (\sim AR_{i,j})}{\text{sum}(N)}$$

Organization Agent Resource Waste Congruence

Go To Measures by Report

Informal Name

None

Citation

Lee, 2004

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

Across all agents, the resources that agents have that are not required to do their assigned tasks.

Input :

Agent by Resource : DataType=binary.

Agent by Task : DataType=binary.

Resource by Task : DataType=binary.

Output :

Graph Level : DataType=real.

$$\Re \in [0,1]$$

As in Agent Waste, let N = AT*RT'

Then

$$\text{Organization Agent Needs} = \frac{\sum_{i=1}^{|A|} \sum_{j=1}^{|R|} AR_{i,j} * (N_{i,j} = 0)}{\text{sum}(AR)}$$

Organization Task Knowledge Needs Congruence

Go To Measures by Report

Informal Name

None

Citation

Lee, 2004

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

Across all tasks, the knowledge that tasks lack expressed as a percentage of the total knowledge needed by all tasks.

> **Example :** An organization needs many skills to meet it's organizational objectives. However, not all of the skills are present within the organizational network. This measure computes a value for this lack of skills as it is distributed across the organization as a whole.

> ### Input :

> Agent by Knowledge : DataType=binary.

> Agent by Task : DataType=binary.

> Knowledge by Task : DataType=binary.

> ### Output :

> Graph Level : dataType=real

> $$\Re \in [0,1]$$

> As in Task Needs, let S = AT'*AK

> Then

$$\text{Organization Task Needs} = \frac{\sum_{i=1}^{|T|} \sum_{j=1}^{|K|} KT^{t}{}_{i,j} *(S_{i,j} = 0)}{\text{sum}(KT)}$$

 Organization Task Knowledge Waste Congruence

Go To Measures by Report

Informal Name

None

Citation

520

Lee, 2004

Minimum and Maximum Values

> **Minimum :** 0

> **Maximum :** 1

Computational Time

Medium

Description

Across all agents, the knowledge that agents have that are not required to do their assigned tasks.

> **Input :**

>> Agent by Knowledge : DataType=binary.

>> Agent by Task : DataType=binary.

>> Knowledge by Task : DataType=binary.

> **Output :**

>> Graph Level : dataType=real

$$\Re \in [0,1]$$

>> As in Task Waste, let S = AT'*AK

>> Then

$$\text{Organization Task Waste} = \frac{\sum_{i=1}^{|T|} \sum_{j=1}^{|K|} S_{i,j} * (\sim KT_{i,j})}{\text{sum}(S)}$$

Organization Task Resource Needs Congruence

Informal Name

None

Citation

Lee, 2004

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

<pMediumFast

Description

Across all tasks, the resources that tasks lack expressed as a percentage of the total resource needed by all tasks.

Example : An organization needs many resources to meet it's organizational objectives. However, not all of the resources are present within the organizational network. This measure computes a value for this lack of resources as it is distributed across the organization as a whole.

Input :

Agent by Resource : DataType=binary.

Agent by Task : DataType=binary.

Resource by Task : DataType=binary.

Output :

Graph Level : dataType=real

$$\Re \in [0,1]$$

As in Task Needs, let S = AT'*AR

Then

$$\text{Organization Task Needs} = \frac{\sum_{i=1}^{|T|} \sum_{j=1}^{|\mathbb{R}|} RT^t_{i,j} * (S_{i,j} = 0)}{\text{sum}(RT)}$$

Organization Task Resource Waste Congruence

Informal Name

None

Citation

Lee, 2004

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

Across all agents, the resources that agents have that are not required to do their assigned tasks.

Input :

Agent by Resource : DataType=binary.

Agent by Task : DataType=binary.

Resource by Task : DataType=binary.

Output :

Graph Level : dataType=real

$$\Re \in [0,1]$$

As in Task Waste, let S = AT'*AR

Then

$$\text{Organization Task Waste} = \frac{\displaystyle\sum_{i=1}^{|T|} \sum_{j=1}^{|R|} S_{i,j} *(\sim RT_{i,j})}{\text{sum}(S)}$$

Organizational Availability

Informal Name

None

Citation

Minimum and Maximum Values

Minimum :

Maximum :

Computational Time

Slow

Description

An organization is available to do a task if it has the required agents and the required resources.Computes the degree to which organizations are

available to do tasks based on their access to resources and roles that are needed to do the tasks.

Input :

Agent by Organization : DataType=binary.

Agent by Task : DataType=binary.

Organization by Resource : DataType=binary.

Resource by Task : DataType=binary.

Output :

Node Level : DataType=real.

Out Degree Centrality

Go To Measures by Report

Informal Name

None

Citation

Wasserman, Stanley and Katherine Faust. "Social Network Analysis: Methods and Applications." Cambridge: Cambridge University Press, 1994.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Fast

Description

For any node, (e.g. an individual or a resource), the out-links are the connections that the node of interest sends to other nodes. For example,

imagine an agent by knowledge matrix then the number of out-links an agent would have is the number of pieces of knowledge it is connected to.

The scientific name of this measure is out-degree and it is calculated on the agent by agent matrices. Individuals or organizations who are high in most knowledge have more expertise or are associated with more types of knowledge than are others. If no sub-network connecting agents to knowledge exists, then this measure will not be calculated. The scientific name of this measure is out degree centrality and it is calculated on agent by knowledge matrices. Individuals or organizations who are high in "most resources" have more resources or are associated with more types of resources than are others. If no sub-network connecting agents to resources exists, then this measure will not be calculated. The scientific name of this measure is out degree centrality and it is calculated on agent by resource matrices.

Input :

Agent by Agent : DataType=real.

Output :

Node Level : DataType=real and Type=row.

$\Re \in [0,1]$ (If N is binary)

Consider the matrix representation X of a network with m rows and n columns. Each row i is assigned an out-degree centrality based on the sum of entries in its row.

The Out-Degree Centrality

$$\text{row i} = \frac{1}{n} \sum_{j=1}^{n} X(i, j)$$

NOTE : This measure is also called Row Degree Centrality.

Out Degree Network Centralization

Go To Measures by Report

Informal Name

None

Citation

NetStat

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Fast

Description

A centralization of a square network based on the Out-Degree Centrality of each node.

Example : In a given organization, all employees have so many links to other employees. This is their out degree connections. This measure computes a value for the entire organization as a whole as to how central these connections are within the organization.

Input :

Square matrix : DataType=real.

Output :

Graph Level : DataType=real.

$$\Re \in [0,1]$$

Let N be a unimodal network with n nodes.

let

$$d_i = \text{Out Degree Centrality of entity i}$$

let

$$\bar{d} = \max\{d_i \mid 1 \leq i \leq n\}$$

527

Then

$$\text{Out Degree Network Centralization} = \left(\sum_{1 \leq i \leq n} \overline{d} - d_i \right) / D$$

where D = (n-2) if N is undirected, and (n-1) otherwise.

Overall Role Based Availability

Go To Measures by Report

Informal Name

None

Citation

Minimum and Maximum Values

   Minimum :

   Maximum :

Computational Time

Slow

Description

Computes the degree to which agents are available to do tasks based on their access to knowledge and resources and roles that are needed to do the tasks.

   Input : Agent by Knowledge matrix with DataType=binary.

   Input : Agent by Resource matrix with DataType=binary.

   Input : Knowledge by Role matrix with DataType=binary.

   Input : Resource by Role matrix with DataType=binary.

Input : Task by Role matrix with DataType=binary.

Input : Task by Role matrix with DataType=binary.

Output : Agent Node Level with DataType=real.

Overall Task Completion

Go To Measures by Report

Informal Name

None

Citation

Carley, Kathleen M., 2002. Summary of Key Network Measures for Characterizing Organizational Architectures. Unpublished Document: CMU 2002

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

The percentage of tasks that can be completed by the agents assigned to them, based solely on whether the agents have the requisite knowledge and resources to do the tasks.

NOTE : Task Completion Overall is a combination of the Task Completion Knowledge and the Task Completion Resource.

Input :

Agent by Knowledge : DataType=binary.

Agent by Resource : DataType=binary.

Agent by Task : DataType=binary.

Knowledge by Task : DataType=binary.

Resource by Task : DataType=binary.

Output :

Graph Level : DataType=real

Performance as Accuracy

Go To Measures by Report

Informal Name

None

Citation

Carley, Kathleen M., 2002. Summary of Key Network Measures for Characterizing Organizational Architectures. Unpublished Document: CMU 2002

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

Measures how accurately agents can perform their assigned tasks based on their access to knowledge and resources.

Input :

Agent by Knowledge DataType=binary.

Agent by Task DataType=binary.

530

Agent by Resource DataType=binary.

Knowledge by Task DataType=binary.

Resource by Task DataType=binary.

Output :

Graph Level : DataType=real

$$\Re \in [0,1]$$

One of the following pairs of matrices must exist: AK/KT, AR/RT. If both exist, then the measure first concatenates them into [AK AR], [KT RT] and uses them.

Accuracy is computed based on the binary classification problem. It is computed in one of two ways:

(1) Knowledge based: Let b be a binary string of length |K|, let N=KT', and let S=AK. Fix a task t.

let

$$\text{answer} = (\sum_{1 \le k \le |K|} N_{t,k} b_k / \sum_{1 \le k \le |K|} N_{t,k} > .5)$$

which is the correct classification of b with respect to task t.

Now, let I={ i | AT(i,t)=1}.

let

$$\text{answer}(i) = (\sum_{1 \le k \le |K|} N_{t,k} S_{i,k} b_k / \sum_{1 \le k \le |K|} N_{t,k} S_{i,k} > .5), i \in I.$$

This is agent i's classification of b with respect to t.

The group of agents classify b using majority voting.

That is,

$$\text{group\_answer} = (\frac{1}{|I|} \sum_{i \in I} \text{answer}(i) > .5).$$

Then, if group_answer = answer, then the group was accurate, otherwise not.

This is repeated multiple times for each task, and across all tasks. The percentage correct is Performance as Accuracy.

(2) Resource based: let N=RT' and S=AR in the analysis of case (1).

Personnel Cost

Informal Name

None

Citation

Ashworth, M. and Carley, Kathleen, M., 2003, "Critical Human Capital, Working Paper, CASOS" Carnegie Mellon, Pittsburgh PA.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Fast

Description

Total number of people reporting to an agent, plus its total knowledge, resources, and tasks.

Personnel Cost is the sum of the in-degree centrality of all AA matrices in the graph (this computes the number of agents reporting to each agent), plus the out-degree centrality of any AK, AR, and AT graphs.

Example : In a given organization, five employees report to Sue. All of the knowledge, resource and tasks connections attributed all the employees and Sue constitute Personnel Cost. This measure then

computes a value for Sue based on all these connections of her employees.

Input :

Agent by Agent : DataType=binary.

Agent by Knowledge : DataType=binary.

Agent by Resource : DataType=binary.

Agent by Task : DataType=binary.

Output :

Node Level : Type=agent and DataType=real.

$$\Re \in [0,1]$$

Personnel Cost is the sum of the in-degree centrality of all AA matrices in the graph (this computes the number of agents reporting to each agent), plus the out-degree centrality of any AK, AR, and AT graphs.

This sum is scaled so that each entity gets a score in [0,1].

Pooled Link Count

Informal Name

None

Citation

Carley, Kathleen M., 2002. Summary of Key Network Measures for Characterizing Organizational Architectures. Unpublished Document: CMU 2002

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Fast

Description

The percentage of pooled links in a unimodal network. A pooled is an link (i,j) such that there exists at least one other link (i,k) in the network, and k <> j.

Example : A pooled link in a network N=(V,E) is an link (i,j)∈ E such that there exists at least one other link (i,k) ∈ E, and k ≠ j.

Input :

Square matrix : DataType=binary.

Output :

Graph Level : DataType=real.

$$\Re \in [0,1]$$

Let M be the adjacency matrix representation of a unimodal network.

Let **S = { (i,j) | M(i,j)=1 ∧ sum(M(:,j))>1 }**

In other words: edge (i,j) is a pooled link iff the in-degree of entity j > 1.

Then Pooled Link Count = |S| / |E|

Potential Boundary Spanner

Go To Measures by Report

Informal Name

None

Citation

Cormen, Leiserson, Rivest, Stein 2001. "Introduction to Algorithms" Second Edition. Cambridge, MA: MIT Press, 2001.

Minimum and Maximum Values

> **Minimum :** 0

> **Maximum :** 1

Computational Time

Slow

Description

Individuals or organizations that connect groups are those that are connecting disconnected groups. The individual or organization may be connected to only one or a few members of each group. This is a composite measure that is high when the agent is potentially influential but is not in the know. This is calculated as the ration of betweenness centrality to total degree centrality and is calculated on agent by agent networks.

Locates nodes that are **high in betweenness centrality, but low in total degree centrality**, and therefore potentially act as links between groups of entities.

Example : This measures finds an individual who could likely have great potential to interact with other parts of an organization based on their exisiting connections.

> **Input :**

> Square matrix : DataType=binary and Directed=No.

> **Output :**

> Node Level : DataType=real

> $$\Re \in [0,1]$$

> For each node in the network, its Potential Boundary Spanner is its Betweenness Centrality divided by its Total Degree Centrality.

Potential Workload

Informal Name

None

Citation

Carley, Kathleen M., 2002. Summary of Key Network Measures for Characterizing Organizational Architectures. Unpublished Document: CMU 2002.

Minimum and Maximum Values

Minimum :

Maximum :

Computational Time

Medium

Description

Maximum knowledge and resources an agent could use to do tasks if it were assigned to all tasks.

Input :

Example : In a given organization, Bob is assigned all task that exist in the organization. This measure would compute a value expressing Bob's potential to carry out all the tasks based on his connections to the resources and knowledge needed for the tasks.

Input :

Agent by Knowledge : DataType=binary.

Agent by Resource : DataType=binary.

Knowledge by Task : DataType=binary.

Resource by Task : DataType=binary.

Output :

Node Level : DataType=real

536

Radials

Informal Name

None

Citation

Carley, Kathleen M., 2002

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Fast

Description

Radial are nodes that have a triad count (the number of triads each node is in) that is less than the mean minus one standard deviation.

Input :

Square matrix : DataType=binary.

Output :

Node Level : DataType=binary.

Reciprocal Link Count

Informal Name

None

Citation

Carley, Kathleen M., 2002. Summary of Key Network Measures for Characterizing Organizational Architectures. Unpublished Document: CMU 2002

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Fast

Description

The percentage of links in a unimodal network that are reciprocated. A link (i,j) in the network is reciprocated if link (j,i) is also in the network.

Input :

Square matrix : DataType=binary.

Output :

Graph Level : DataType=real

$$\Re \in [0,1]$$

Let G=(V,E) be the graph representation of a network.

Let S=card{ (i,j)∈E|i<,(j,i)∈E}

Then **Reciprocral Link Count = |S| / |E|**

Relative Cognitive Distinctiveness

Informal Name

None

Citation

Carley, Kathleen M., 2002

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Slow

Description

Measures the degree to which each pair of agents has complementary knowledge, expressed as the percent of total knowledge. Each value is normalized by the row sum.

Input :

Agent by Knowledge : DataType=binary.

Output :

Dyad Level : Agent by Agent and DataType=real.

For each pair of agents (i,j) compute the number of knowledge bits they have exactly opposite. Then normalize this sum by the total number of knowledge bits. In effect, this is the exclusive-OR of their knowledge vectors.

$$CD_{i,j} = \frac{\sum_{k=1}^{|K|} (AK_{i,k} * \sim AK_{j,k}) + (\sim AK_{i,k} * AK_{j,k})}{|K|}$$

$$CD_{i,i} = 0$$

NOTE : Note that the CD output matrix is symmetric.

Relative Cognitive Distinctiveness normalizes each element of CD as follows:

$$RCD_{i,j} = \frac{CD_{i,j}}{\sum_{j=1}^{|K|} CD_{i,j}}$$

Thus, the elements of the ith row are normalized by the ith row sum.

Relative Cognitive Expertise

Go To Measures by Report

Informal Name

None

Citation

Carley, Kathleen M., 2002

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Slow

Description

Measures the degree to which each pair of agents has complementary knowledge, expressed as a percentage of the knowledge of the first agent. Each value is normalized by the row sum. Individuals or organizations that are high in specialized knowledge are those that have expertise or are connected to types of knowledge that few others have.

Input :

Agent by Knowledge : DataType=binary.

Output :

Node Level : Type=agent and DataType=real.

Dyad Level : Agent by Agent and DataType=real.

For each pair of agents (i,j) compute the number of knowledge bits that j knows that i does not know. Then normalize this sum by the total number of knowledge bits that agent i does not know.

$$CE_{i,j} = \frac{\sum_{k=1}^{|K|}(\sim AK_{i,k} * AK_{j,k})}{(|K| - \sum_{k=1}^{|K|} AK_{i,k})}$$

$$CE_{i,i} = 0$$

NOTE : Note that the CD output matrix is NOT-symmetric.

Relative Cognitive Expertise normalizes each element of CE as follows:

$$RCE_{i,j} = \frac{CE_{i,j}}{\sum_{j=1}^{|K|} CE_{i,j}}$$

Thus, the elements of the ith row are normalized by the ith row sum.

Relative Cognitive Resemblance

Go To Measures by Report

Informal Name

None

Citation

Carley, Kathleen M., 2002

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Slow

Description

Measures the degree to which each pair of agents has the exact same knowledge. Each value is normalized by the row sum.

Input :

Agent by Knowledge : DataType=binary.

Output :

Dyad Level : Agent by Agent and DataType=real.

$$\Re \in [0,1]$$

For each pair of agents (i,j) compute the number of knowledge bits they have in common - whether known or unknown. Then normalize this sum by the total number of knowledge bits.

$$CR_{i,j} = \frac{\sum_{k=1}^{|K|} (AK_{i,k} * AK_{j,k}) + (\sim AK_{i,k} * \sim AK_{j,k})}{|K|}$$

$$CR_{i,i} = 1$$

NOTE : The CR output matrix is symmetric.

Relative Cognitive Resemblance normalizes each element of CR as follows:

$$RCR_{i,j} = \frac{CR_{i,j}}{\sum\limits_{j=1}^{|K|} CR_{i,j}}$$

Thus, the elements of the i**th** row are normalized by the i**th** row sum.

Relative Cognitive Similarity

Go To Measures by Report

Informal Name

None

Citation

Carley, Kathleen M., 2002

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Slow

Description

Measures the degree to which each pair of agents have overlapping knowledge. Each value is normalized by the row sum.

Input :

Agent by Knowledge and DataType=binary.

Output :

$$\Re \in [0,1]$$

Node Level : Type=agent and DataType=real.

Dyad Level : Agent by Agent and DataType=real.

For each pair of agents (i,j) compute the number of knowledge bits they have in common. Then normalize this sum by the total knowledge between them.

$$CS_{i,j} = \frac{\sum_{k=1}^{|K|}(AK_{i,k} * AK_{j,k})}{\sum_{k=1}^{|K|}(AK_{i,k} + AK_{j,k})} |$$

$$CS_{i,i} = 1$$

NOTE : The CS output matrix is symmetric.

Relative Cognitive Similarity normalizes each element of CS as follows:

$$RCS_{i,j} = \frac{CS_{i,j}}{\sum_{j=1}^{|K|} CS_{i,j}}$$

Thus, the elements of the i**th** row are normalized by the i**th** row sum.

Relative Expertise

Go To Measures by Report

Informal Name

None

Citation

Carley, Kathleen M., 2002

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

The degree of dissimilarity between agents based on shared knowledge. Each agent computes to what degree the other agents know what they do not know. Individuals or organizations that are high in specialized knowledge are those that have expertise or are connected to types of knowledge that few others have. The scientific name of this measure is relatively unique and it is calculated on agent by knowledge matrices.

Input :

Agent by Knowledge : DataType=binary.

Output :

Node Level : Type=agent and DataTpe=real.

Relative Similarity

Go To Measures by Report

Informal Name

None

Citation

Carley, Kathleen M., 2002

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

The degree of similarity between two agents based on shared knowledge. Each agent computes to what degree the other agents know what they know.

Input :

Agent by Knowledge : DataType=binary.

Output :

Node Level : Type=agent and DataType=real.

Resemblance Correlation

Go To Measures by Report

Informal Name

None

Citation

Carley, Kathleen, M., 2002

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Slow

Description

Measures the degree to which each pair of rows has the exact same data.

Input :

DataType=binary.

Output :

Node Level : Type=row and DataType=real.

Dyad Level : Row by Row and DataType=real.

Resource Actual Workload

Informal Name

None

Citation

Carley, Kathleen M., 2002. "Summary of Key Network Measures for Characterizing Organizational Architectures." Unpublished Document: CMU 2002

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

The resources an agent uses to perform the tasks to which it is assigned.

Input :

Agent by Resource : DataType=binary

Resource by Task : DataType=binary

Agent by Task : with DataType=binary

Output :

Node Level : Type=agent and DataType=real.

Resource Based Access Index

Informal Name

None

Citation

Ashworth, M. and Carley, Kathleen M., 2003, "Critical Human Capital", Working Paper, CASOS, Carnegie Mellon, Pittsburgh PA.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Fast

Description

Boolean value which is true if an agent has exclusive access to a resource and who is known by exactly one other agent. The one agent known also has its KAI set to one.

An node level measure that computes the average shortest path length between entities, excluding infinite distances. This is known as the Resource Access Index (RAI) value.

Example : Let us imagine a social network where 50 employees have many connections to each other. The average number of connections per employee is the Access Index.

Input :

548

Agent by Agent : DataType=binary

Agent by Resource : DataType=binary.

Output :

Node Level : Type=agent and DataType=binary.

The Resource Access Index (RAI) for agent i is defined as follows:

$$S_i = \{s \mid AR(i,s) \wedge \big(\text{sum}(AR(:,s)) = 1\big) \wedge \big(\text{sum}(A(i,:)) = 1\big)\}$$

let

Then

$$RAI_i = \big((S_i \neq \varnothing) \vee \big(\exists\, j \mid S_j \neq \varnothing \wedge A(j,i) = 1\big)\big)$$

Resource Based Task Completion

Go To Measures by Report

Informal Name

None

Citation

Carley, Kathleen M., 2002. Summary of Key Network Measures for Characterizing Organizational Architectures. Unpublished Document: CMU 2002

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

The percentage of tasks that can be completed by the agents assigned to them, based solely on whether the agents have the requisite resources to do the tasks.

> **Example :** Bob is assigned 10 Tasks to complete which require a certain amount of resources to finish. However, on average only 10-percent of those resources will be available to Bob for him to complete the tasks given. This measure would compute a value for the likelyhood that Bob would be able to complete any given task.

NOTE : Task Completion Overall is a combination of the Task Completion Knowledge and the Task Completion Resource.

Input :

Agent by Resource : dataType=binary.

Agent by Task : dataType=binary.

Resource by Task : dataType=binary.

Output :

Graph Level : DataType=real

Find the tasks that cannot be completed because the agents assigned to the tasks lack necessary knowledge.

let Need = [(AT'*AR) - KT']

let S = { i | , j : Need(i,j) < 0 }

Knowledge Based Task Completion is the percentage of tasks that could be completed = (|T|-|S|) / |T|

Resource Diversity

Go To Measures by Report

Informal Name

Who Has What

Citation

Borgatti, S.P. 2003. The Key Player Problem. Dynamic Social Network Modeling and Analysis: Workshop Summary and Papers, R.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Fast

Description

The distribution of difference in resource sharing. This is the Herfindahl-Hirshman index applied to column sums of AR.

Example : Characterizes the network in terms of how the items in a resource network **(who has what)** are non-uniformly distributed across employees

Input :

Agent by Resource : DataType=binary.

Output :

Graph Level : DataType=real

$$\Re \in [0,1]$$

This is the Herfindahl-Hirshman index (economics: sum of the squares of each firm's market share) applied to the normalized column sums of AK. This measures the degree to which knowledge is equally known.

let

$$w_k = \sum_{i=1}^{|A|} AK(i,k) \text{ , for } 1 \le k \le |K|$$

let

$$W = \sum_{k=1}^{|K|} w_k$$

Then

$$\text{Knowledge Diversity} = 1 - \sum_{k=1}^{|K|} \left(w_k / W \right)^2$$

Resource Exclusivity

Informal Name

None

Citation

Ashworth, 2002

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Fast

Description

Detects agents who have singular resource access.

Example : A key resource that supports an entire organization is only accessible through one person.

Input :

Agent by Resource : DataType=binary.

Node Level : DataType=real and Type=agent.

$$\Re \in [0,1]$$

The **Resource Exclusivity Index (REI)** for agent i is defined as follows:

$$\sum_{j=1}^{|K|} AR(i, j) * \exp(1 - \text{sum}(AR(:, j)))$$

The **Resource Exclusivity Index** (REI) for agent i is defined exactly as for Resource Based Exclusivity.

Resource Load

Informal Name

None

Citation

Carley, Kathleen M., 2002. "Summary of Key Network Measures for Characterizing Organizational Architectures." Unpublished Document: CMU 2002.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Fast

Description

Average number of resources per agent.

Input :

Agent by Resource : DataType=binary

Graph Level : DataType=real.

🔲 Resource Negotiation

Go To Measures by Report

Informal Name

None

Citation

Carley, Kathleen M., 2002. "Summary of Key Network Measures for Characterizing Organizational Architectures." Unpublished Document: CMU 2002

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

The extent to which agents need to negotiate with each other because they lack the resources to complete their assigned tasks.

Example : In a given organization, Bob does not have access to the resources needed to complete a certain task but Sue does have it. Bob must negotiate with Sue to gain those resources. The extent to which others in the organization must negotiate for resources that others have is the organization's resource negotiation value.

Input :

Agent by Tasks : DataType=binary.

Agent by Resource : DataType=binary.

Resource by Tasks : DataType=binary.

Output :

Graph Level : DataType=real.

Resource Omega

Go To Measures by Report

Informal Name

None

Citation

Carley, Kathleen M., Dekker, D., Krackhardt, D (2000). How Do Social Networks Affect Organizational Knowledge Utilitization.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

The degree to which agents reuse resources while doing their tasks.

Example : Sue has X amount of tasks within her company. How many of those tasks that require a resource that can be used for one or more of her other tasks is omega knowledge. This measure computes a value for such relations to the resources at the network level.

Input :

Agent by Task : DataType=binary.

Resource by Task : DataType=binary.

Task by Task : DataType=binary.

Output :

Graph Level : DataType=real.

Let TAT = TA*TA'

Let N = ((TT'@TAT)*KT')@KT'

![icon] Resource Potential Workload

Go To Measures by Report

Informal Name

None

Citation

Carley, Kathleen M., 2002. Summary of Key Network Measures for Characterizing Organizational Architectures. Unpublished Document: CMU 2002.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

Maximum resources an agent could use to do tasks if it were assigned to all tasks.

Example : In a given organization, let us say Bob is assigned all task that exist in the organization. This measure would compute a value

expressing Bob's potential to carry out all the tasks based on his connections to the resources and knowledge needed for the tasks.

Input :

Agent by Resource : DataType=binary.

Resource by Task : DataType=binary.

Output :

Node Level : Type=agent and DataType=real.

Potential Knowledge Workload for agent i = sum((AK*KT)(i,:))/sum(KT)

Resource Rendundancy

Go To Measures by Report

Informal Name

None

Citation

Carley, Kathleen M., 2002. "Summary of Key Network Measures for Characterizing Organizational Architectures." Unpublished Document: CMU 2002.

Minimum and Maximum Values

Minimum : 0

Maximum : $[(|R|-1)*|T|]$

Computational Time

Fast

Description

Average number of redundant agents per resource. An agent is redundant if there is already an agent that has the resource.

Agent by Resource : DataType=binary.

Graph Level : DataType=real.

$$\Re \in [0,(|R|-1)*|T|]$$

This is the Column Redundancy of matrix RT.

Resource Under Supply

Go To Measures by Report

Informal Name

None

Citation

Carley, Kathleen M., 2002. Summary of Key Network Measures for Characterizing Organizational Architectures. Unpublished Document: CMU 2002

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

The extent to which the resources needed to do tasks are unavailable in the entire organization.

Example : In an organization where Bob needs to complete Task A, he needs access to Resource B, which in this case might be a computer.

However, what would happen if Resource B were simply unavailable? Say the computer was offline 50-percent of the time. This computer would likely under supply the task of completing Task A. This measure would compute a value to the effect that this under supply factor affected the ability to perform the task at hand.

Input :

Agent by Task : dataType=binary.

Agent by Resource : dataType=binary.

Resource by Task : dataType=binary.

Output :

Graph Level : DataType=real

This is the number of required knowledge not available to tasks based on the current agent task assignments normalized by the number of tasks.

Resource Under Supply = sum(dich(RT - AR'*RT)) / |T|

Role Based Knowledge Availability

Go To Measures by Report

Informal Name

None

Citation

Behrman, Robert; forthcoming; "Network Analysis of the Structure and Capacity of Brigade Level Military Organizations"; Chapter 3; PhD Thesis

Minimum and Maximum Values

Minimum :

Maximum :

Computational Time

Slow

Description

Computes the number of roles that an agent is qualified to have based on knowledge requirements.

Agent by Knowledge x Knowledge by Role = Agent by Role$_{resource}$
subject to the additional constraint:

> If **Max(Knowledge by Role$j$)=0** (i.e. role j requires no knowledge) then **Agent by Role$_j$**=1

A link **Agent by Role$_{ij}$** indicates that agent i has no access to the knowledge needed for role j.

Input :

Agent by Knowledge : DataType=binary.

Knowledge by Role : DataType=binary.

Output :

Agent by Role : DataType=binary.

Role Based Resource Availability

Go To Measures by Report

Informal Name

None

Citation

Behrman, Robert; forthcoming; "Network Analysis of the Structure and Capacity of Brigade Level Military Organizations"; Chapter 3; PhD Thesis

Minimum and Maximum Values

Minimum :

Maximum :

Computational Time

Slow

Description

Computes the number of roles that an agent is qualified to have based on Resource requirements.

Agent by Resource x Resource by Role = Agent by Role$_{resource}$

A Link **Agent by Role$_{ij}$** indicates that agent i has the skills to use the resource required by role j.

Input :

    Agent by Resource : DataType=binary.

    Resource by Role : DataType=binary.

Output :

    Agent by Role : DataType=binary.

Role Based Knowledge

Go To Measures by Report

Informal Name

None

Citation

Minimum and Maximum Values

Minimum :

Maximum :

Computational Time

Slow

Description

Computes the degree to which agents are available to do tasks based on their access to knowledge and roles that are needed to do the tasks.

Input :

Agent by Knowledge : DataType=binary.

Knowledge by Role : with DataType=binary.

Task by Role matrix : DataType=binary.

Output :

Node Level : Type=agent and DataType=real.

 # Role Resource Availability

Go To Measures by Report

Informal Name

None

Citation

Minimum and Maximum Values

Minimum :

Maximum :

Computational Time

Slow

Description

562

Computes the number of roles that an agent is qualified to have based on resource requirements.

Input : Agent by Resource matrix with DataType=binary.

Input : Resource by Role matrix with DataType=binary.

Output : Agent Node Level with DataType=real.

Row Breadth

Informal Name

None

Citation

Carley, Kathleen M., 2002

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Fast

Description

The fraction of entities with row nodes with degree greater than one.

Example : In a given organization, we have the following entities: employees (agents), computers (resources), benchmarks (tasks). This measure tells us on how many links greater than one exist within these entity types.

Input :

DataType=binary.

Output :

Graph Level : DataType=real.

$$\Re \in [0,1]$$

Let d = the Row degree vector of the network N

Then, the Row Breadth of N is:

$$\frac{\sum_{i=1}^{|d|}(d_i > 1)}{|d|}$$

NOTE : Column breadth is identical, except that d is defined using the column degree, rather than the row degree.

Row Count

Informal Name

None

Citation

Carley, Kathleen M., 2002

Minimum and Maximum Values

Minimum : 0

Maximum : Number of rows in the network

Computational Time

Fast

Description

The number of row nodes in a network.

Input :

DataType=binary.

Output :

Graph Level : DataType=integer.

Result: $Z \geq 0$

Row Degree Centrality

Informal Name

see **Out Degree Centrality**

Row Degree Network Centralization

Informal Name

None

Citation

NetStat

Minimum and Maximum Values

Minimum :

Maximum :

Computational Time

Fast

Description

A centralization based on the degree of the row nodes of a network.

Input :

DataType=real

Output :

Graph Level : Type=row and DataType=real.

$$\Re \in [0,1]$$

Let N be a network with n row nodes.

let

$$d_j = \text{degree of row node } j, \ 1 \le j \le n$$

let

$$\overline{d} = \max\{d_j \mid 1 \le j \le n\}$$

Then

$$\text{Row Degree Network Centralization} = \left(\sum_{1 \le j \le n} \overline{d} - d_j \right) / (n)$$

Row Redundancy

Go To Measures by Report

Informal Name

None

Citation

Carley, Kathleen M., 2002. Summary of Key Network Measures for Characterizing Organizational Architectures. Unpublished Document: CMU 2002

Minimum and Maximum Values

Minimum : 0

Maximum : [(N-1)*M]

Computational Time

Fast

Description

The mean number of row node links in excess of one.

Example : Let us say in our organizational matrix, we have Bob who multiple links to Sue. Let us say this is multiple linkages has an average of 2.5 per employee to the same manager Sue or another. The measure computes a value for this average which is considering row redundancy.

Input :

DataType=binary.

Output :

Graph Level : DataType=real.

$$\Re \in \left[0,(n-1)*m\right]$$ for N dimension m x n

Let M be the matrix representation for a network N of dimension m x n.

let

$$d_i = \max\{0, \operatorname{sum}(M(j,:)) - 1\}$$

for

$$1 \le i \le m;$$

this is the number of column entries in excess of one for row i.

Then

Sequential Link Count

Informal Name

None

Citation

Carley, Kathleen M., 2002. Summary of Key Network Measures for Characterizing Organizational Architectures. Unpublished Document: CMU 2002.

Minimum and Maximum Values

> **Minimum :** 0

> **Maximum :** 1

Computational Time

Fast

Description

The percentage of links in a network that are neither Reciprocal Links nor Pooled Links. Note that an link can be both a Pooled and a Reciprocal link.

Let **G=(V,E)** be the graph representation of a network, and let **X = set of Pooled edges of G**, and **let Y = set of Reciprocal edges of G**.

Then **Sequential Edge Count = | E-X-Y| / |E|**

Example : The percentage of edges in a network that are neither Reciprocal Edges nor Pooled Edges. Note that an edge can be both a Pooled and a Reciprocal edge. Self-loops are ignored.

> **Input :**

Square matrix : DataType=binary.

Output :

Graph Level : DataType=real.

$$\Re \in [0,1]$$

Let **G=(V,E)** be the graph representation of a network, and let **X = set of Pooled edges of G**, and **let Y = set of Reciprocal edges of G**.

Then **Sequential Edge Count = | E-X-Y| / |E|**

Shared Situation Awareness

Go To Measures by Report

Informal Name

None

Citation

Graham, 2004

Minimum and Maximum Values

Minimum : 0

Maximum : Unscaled real number

Computational Time

Medium

Description

Individuals or organizations that are high in group awareness are those that by virtue of their connections to others, what resources they use, what knowledge there is, what tasks there are - have a better understanding of what others are doing.

Example : If we take any given pair of actors in an organizatin, this measure computes a value for how similar they are to each other based on their relationships (i.e., links) to others in the same organization.

Input :

Agent by Agent : DataType=binary.

Output :

Node Level : Type=agent and DataType=real.

$$\Re \text{ , not scaled}$$

Dyad Level : agent by agent and DataType=real.

$$\Re \text{ , not scaled}$$

Formula :

Let

$$\alpha, \beta, \gamma, \delta, \mu \text{ be Real number (these are weights).}$$

Let A = Agent x Agent interaction/communication network.

Let P = Agent x Agent physical proximity network.

Let S = Agent x Agent social demographic similarity network.

Let e = eigenvector centrality measure computed on A.

Let G = geodesics between agents computed on A.

Then the SSA measure between agents i and j is:

Similarity Correlation

Go To Measures by Report

Informal Name

None

Citation

Carley, Kathleen, M., 2002

Minimum and Maximum Values

>**Minimum :** 0

>**Maximum :** 1

Computational Time

Slow

Description

Measures the degree to which each pair of rows have overlapping data.

>**Input :**

>>Matrix : DataType=binary.

>**Output :**

>>Node Level : Type=row and DataType=real.

>>Dyad Level : Row by Row, and DataType=real.

Simmelian Ties

Go To Measures by Report

Informal Name

None

Citation

Krackhardt, David. 1998. "Simmelian Tie: Super Strong and Sticky." In Power and Influence in Organizations., edited by Roderick M. Kramer and Margaret Neale, pp. 21/38. Thousand Oaks, CA: Sage.

Simmel, Georg. 1950. "The Sociology of Georg Simmel." New York: Free Press, 1st free press paperback edition.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

The normalized number of Simmelian ties of each node.

Example : Described informally as ties embedded in cliques and are often associated with **brokers** inside such cliques such that if **Bob** and **Susan** only know of each other becuase of **Chan** and now all of them, Bob, Susan and Chan, now know each other. Chan, Bob, Susan now have **simmelian ties** to each other.

Input :

Square matrix : DataType=binary.

Output :

Node Level : dataType=real.

$$\Re \in [0,1]$$

let N be a square network with n entities. The Simmelian Tie value for entity i is computed as follows:

let NS = network N symmetrized using the Intersect (Minimum) method.

All cliques of size >= 3 are computed on NS and clique membership is recorded. Let c_i be the number of entities that are in a clique with entity i.

Then the Simmelian Ties value for emtity i is c_i.

Skip Link Count

Informal Name

None

Citation

Carley, Kathleen M., 2002. Summary of Key Network Measures for Characterizing Organizational Architectures. Unpublished Document: CMU 2002.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

The fraction of links in a unimodal network that skip levels. An link (i,j) is a skip link if there is a path from node i to node j even after the link (i,j) is removed.

Example : The fraction of edges in a unimodal network that skip levels. An edge (i,j) is a skip edge if there is a path from entity i to entity j even after the edge (i,j) is removed.

Input :

Square matrix : DataType=binary.

Output :

Graph Level : DataType=real.

$$\Re \in [0,1]$$

A skip link in a network **G=(V,E)** is an link (i,j)$\in$E such that j is reachable from i in the **graph G'=(V,E\(i,j))**, that is, the graph G with link (i,j) removed. Skip Count is simply the number of such links in G normalized to be in [0,1] by dividing by |E|.

573

Social Technical Congruence

Informal Name

None

Citation

Cataldo et al, 2006

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

Social technical congruence represents the proportion of coordination requirements that were satisfied through some type of actual coordination activity carried out through some means of communication.

A representation of the coordination requirements among the different workers is obtained by multiplying AT, TT and transpose of AT matrices. This product results in a people by people matrix where a cell ij indicates the extent to which person i works on tasks that share dependencies with the tasks worked on by person j. In other words, the resulting matrix represents the Coordination Requirements or the extent to which each pair of people needs to coordinate their work.

Formally, the Coordination Requirements (CR) matrix is determined by the following product:

Let, **CR = AT * TT * transpose(AT)**

Given a particular CR matrix constructed from relating product dependencies to work dependencies, we can compare it to an Actual

Coordination (CA) matrix that represents the interactions workers engaged in through different means of coordination.

Diff (CR, CA) = card { diff$_{ij}$ | crij > 0 & caij > 0 }

|CR| = card { crij > 0 }

We have,

Congruence(CR, CA) = Diff (CR, CA) / |CR|

Agent by Agent : DataType=binary

Agent by Task : DataType=binary

Task by Task : DataType=binary

Output :

Graph Level : DataType=real.

Dyad Level with Agent (source) by Agent (target) with DataType=real.

$$\Re \in [0,1]$$

The Coordination Requirements (CR) matrix is determined by the following product:

Let, **CR = AT * TT * transpose(AT)**

Given a particular CR matrix constructed from relating product dependencies to work dependencies, we can compare it to an Actual Coordination (CA) matrix that represents the interactions workers engaged in through different means of coordination.

Diff (CR, CA) = card { diff$_{ij}$ | crij > 0 & caij > 0 }

|CR| = card { crij > 0 }

We have,

Congruence(CR, CA) = Diff (CR, CA) / |CR|

Go To Measures by Report

Informal Name

None

Citation

Carley, Kathleen M., 2002. Summary of Key Network Measures for Characterizing Organizational Architectures. Unpublished Document: CMU 2002

Minimum and Maximum Values

Minimum : 0

Maximum : $\in \Re$ [0,|V|-1]

Computational Time

Fast

Description

The average number of out links per node with non-zero out degrees.

Example : Bob has 5 out degree links to 5 other personnel in his company and of those 5 personnel, the average non-zero (i.e., people that make up that personnel that have out-degree links of there own) is 4. Whereas Sue might have 5 personnel out degree links, but those personnel have 3 out-degree links themselves on average. It would figure that Sue might have a greater span of control, since the people she connects do, don't have as many connections as those that Bob connects to does. This measure would compute a value for such relationships called Span of Control.

Input :

Square matrix : DataType=binary.

Output :

Graph Level : dataType=real.

$$\mathfrak{R} \in [0, |V| - 1]$$

let S = set of entities in V that have positive out-degree

let

$$K = \sum_{i \in S} \text{outDegree}(i)$$

Then

$$\text{Span of Control} = K / |S|$$

Strict Knowledge Congruence

Informal Name

None

Citation

Carley, Kathleen M., 2002. Summary of Key Network Measures for Characterizing Organizational Architectures. Unpublished Document: CMU 2002.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

Measures the similarity between what knowledge are assigned to tasks via agents, and what knowledge are required to do tasks. Perfect congruence

occurs when agents have access to knowledge when and only when it is needful to complete tasks.

Knowledge Congruence = 1 iff agents have knowledge when and only when it is needful to complete their tasks. Thus, we compute the knowledge assigned to tasks via agents, and compare it with the knowledge needed for tasks.

Input :

Agent by Knowledge : DataType=binary.

Agent by Task : DataType=binary.

Knowledge by Task : DataType=binary.

Output :

Graph Level : DataType=real.

Knowledge Congruence = 1 iff agents have knowledge when and only when it is needful to complete their tasks. Thus, we compute the knowledge assigned to tasks via agents, and compare it with the knowledge needed for tasks.

let **KAT = dich(AK'*AT)**

for Resource KAT becomes RAT and replace AK with AR

let

$$d = \sum_{i=1}^{|K|} \sum_{j}^{|T|} KAT(i, j) * KT(i, j)$$

for Resource replace KT with RT

let <**d = d / (|K|*|T|)**, which normalizes d to be in [0,1]

for Resource replace |K| with |R|

Then Knowledge Congruence = 1 - d

Strict Resource Congruence

Informal Name

None

Citation

Carley, Kathleen M., 2002. Summary of Key Network Measures for Characterizing Organizational Architectures. Unpublished Document: CMU 2002.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

Measures the similarity between resources assigned to tasks via agents, and what resources are required to do tasks. Perfect congruence occurs when agents have resources when and only when it is needful to complete tasks.

Knowledge Congruence = 1 iff agents have knowledge when and only when it is needful to complete their tasks. Thus, we compute the knowledge assigned to tasks via agents, and compare it with the knowledge needed for tasks.

Input :

Agent by Resource : DataType=binary.

Agent by Task : DataType=binary.

Resource by Task : DataType=binary.

Output :

Graph Level : DataType=real.

Resource Congruence = 1 iff agents have knowledge when and only when it is needful to complete their tasks. Thus, we compute the resource assigned to tasks via agents, and compare it with the resource needed for tasks.

let **RAT = dich(AK'*AT)**

   for Knowledge RAT becomes KAT and replace AR with AK

let

$$d = \sum_{i=1}^{|R|} \sum_{j}^{|T|} RAT(i, j) * RT(i, j)$$

   for Knowledge replace RT with KT

   let

Strong Component Count

Go To Measures by Report

Informal Name

None

Citation

Wasserman, Stanley and Katherine Faust. "Social Network Analysis: Methods and Applications." Cambridge: Cambridge University Press, 1994.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

The number of strongly connected components in a network. The network can be directed or undirected.

Given a square network G=(V,E), the Strong Component Count is the number of strongly connected components in G. This is computed directly on G, whether or not G is directed.

Input :

Square matrix : DataType=binary.

Output :

Graph Level : dataType=integer.

$$Z \in [0, |V|]$$

Given a square, symmetric network **G=(V,E)**, the Strong Component Count is the number of strongly connected components in G. This is computed directly on G, whether or not G is directed.

Task Exclusivity

Go To Measures by Report

Informal Name

None

Citation

Ashworth, 2002

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Fast

Description

Detects agents who exclusively perform tasks. Individuals or organizations that are high in complete exclusivity for tasks are those that are engaged in or assigned to tasks or activities for which no one else is engaged in or assigned to. The scientific name of this measure is complete exclusivity and it is calculated on agent by task matrices.

Example : A key task that supports an entire organization is only accessible through one person.

Input :

Agent by Task : DataType=binary.

Output :

Node Level : Type=agent and DataType=real

$$\Re \in [0,1]$$

The **Task Exclusivity Index (KEI)** for agent i is defined as follows:

$$\sum_{j=1}^{|K|} AT(i,j) * \exp(1 - sum(AT(:,j)))$$

The **Task Exclusivity Index** (TEI) for agent i is defined exactly as for Task Based Exclusivity.

d = d / (|K|*|T|), which normalizes d to be in [0,1]

for Knowledge replace |R| with |K|

Then Resource Congruence = 1 - d

Task Knowledge Needs Congruence

Go To Measures by Report

Informal Name

None

Citation

Lee, 2004

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

The number of knowledge not supplied to a task, but required to do the task, expressed as a percentage of the total knowledge required for the task.

Task Knowledge Needs compares the knowledge requirements of each task with the knowledge available to the task via agents assigned to it. It is similar to Knowledge Congruence, but quantifies only the under supply of knowledge to tasks.

Input :

Agent by Knowledge : DataType=binary.

Agent by Task : DataType=binary.

Knowledge by Task : DataType=binary.

Output :

Node Level : Type=Task and DataType=real

$$\Re \in [0,1]$$

Task Knowledge Needs compares the knowledge requirements of each task with the knowledge available to the task via agents assigned to it. It is similar to Knowledge Congruence, but quantifies only the under supply of knowledge to tasks.

Let **S = AT'*AK = knowledge** supplied to tasks via assigned agents

We need to sum the knowledge required but not supplied.

Thus,

$$\text{Task Knowledge Needs for task i} = \frac{\sum_{j=1}^{|K|} KT^t_{i,j} * (S_{i,j} = 0)}{\sum_{j=1}^{|K|} KT^t_{i,j}}$$

The equation for Agent Resource Needs replaces AK with AR, and KT with RT.

Task Knowledge Waste Congruence

Go To Measures by Report

Informal Name

None

Citation

Lee, 2004

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

The number of knowledge supplied to a task via agents that are not required by it expressed as a percentage of the total knowledge required for the task.

Task Knowledge Waste compares the knowledge requirements of each task with the knowledge available to the task via agents assigned to it. It is

similar to Knowledge Congruence, but quantifies only the over supply of knowledge to tasks.

Input :

Agent by Knowledge : DataType=binary.

Agent by Task : DataType=binary.

Knowledge by Task : DataType=binary.

Output :

Node Level : Type=task and DataType=real.

$$\Re \in [0,1]$$

Task Knowledge Waste compares the knowledge requirements of each task with the knowledge available to the task via agents assigned to it. It is similar to Knowledge Congruence, but quantifies only the over supply of knowledge to tasks.

Let **S = AT'*AK = knowledge** supplied to tasks via assigned agents

We need to sum the knowledge supplied but not required.

Thus,

$$\text{Task Knowledge Waste for task i} = \frac{\sum_{j=1}^{|K|} S_{i,j} * (\sim KT^{t}_{i,j})}{\sum_{j=1}^{|K|} S_{i,j}}$$

The equation for Agent Resource Waste replaces AK with AR, and KT with RT.

Task Resource Needs Congruence

Go To Measures by Report

Informal Name

None

Citation

Lee, 2004

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

The number of resources not supplied to a task, but required to do the task, expressed as a percentage of the total resources required for the task.

Task Knowledge Needs compares the knowledge requirements of each task with the knowledge available to the task via agents assigned to it. It is similar to Knowledge Congruence, but quantifies only the under supply of knowledge to tasks.

Input :

Agent by Resource : DataType=binary.

Agent by Task : DataType=binary.

Resource by Task : DataType=binary.

Output :

Node Level : Type=task DataType=real.

$$\Re \in [0,1]$$

Task Knowledge Needs compares the knowledge requirements of each task with the knowledge available to the task via agents assigned to it. It is similar to Knowledge Congruence, but quantifies only the under supply of knowledge to tasks.

Let **S = AT'*AR = resource** supplied to tasks via assigned agents

We need to sum the resources required but not supplied.

Thus,

$$\text{Task Resource Needs for task i} = \frac{\sum_{j=1}^{|R|} RT^t_{i,j} * (S_{i,j} = 0)}{\sum_{j=1}^{|R|} RT^t_{i,j}}$$

The equation for Agent Resource Needs replaces AR with AK, and RT with KT.

Task Resource Waste Congruence

Informal Name

None

Citation

Lee, 2004

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

The number of resources supplied to a task via agents that are not required by it expressed as a percentage of the total resources required for the task.

Task Knowledge Waste compares the knowledge requirements of each task with the knowledge available to the task via agents assigned to it. It is similar to Knowledge Congruence, but quantifies only the over supply of knowledge to tasks.

Input :

Agent by Resource : DataType=binary.

Agent by Task ; DataType=binary.

Resource by Task : DataType=binary.

Output :

Node Level : Type=task and DataType=real.

$$\Re \in [0,1]$$

Task Knowledge Needs compares the knowledge requirements of each task with the knowledge available to the task via agents assigned to it. It is similar to Knowledge Congruence, but quantifies only the under supply of knowledge to tasks.

Let **S = AT'*AK = knowledge** supplied to tasks via assigned agents

We need to sum the knowledge required but not supplied.

Thus,

$$\text{Task Resource Waste for task } i = \frac{\sum_{j=1}^{|R|} S_{i,j} * (\sim RT^{t}{}_{i,j})}{\sum_{j=1}^{|R|} S_{i,j}}$$

The equation for Agent Resource Needs replaces AK with AR, and KT with RT.

Total Degree Centrality

Informal Name

In The Know

Citation

Wasserman, Stanley and Katherine Faust. "Social Network Analysis: Methods and Applications." Cambridge: Cambridge University Press, 1994.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Fast

Description

The Total Degree Centrality of a node is the normalized sum of its row and column degrees. Individuals or organizations who are "in the know" are those who are linked to many others and so, by virtue of their position have access to the ideas, thoughts, beliefs of many others. Individuals who are "in the know" are identified by degree centrality in the relevant social network. Those who are ranked high on this metrics have more connections to others in the same network. The scientific name of this measure is total degree centrality and it is calculated on the agent by agent matrices.

Individuals or organizations who are "in the know" are those who are linked to many others and so, by virtue of their position have access to the ideas, thoughts, beliefs of many others. Individuals who are "in the know" are identified by degree centrality in the relevant social network.

Input :

Square matrix : DataType=real.

Output :

Node Level : DataType=real.

$$\Re \in [0,1]$$

Consider the matrix representation X of a square network with n entities. Each entity is assigned a value based on the sum of its row and columns.

The Total Degree Centrality for

$$\text{entity i} = \frac{1}{2(n-1)} \sum_{i=1}^{n} \sum_{\substack{j=1 \\ i \neq j}}^{n} X(i, j)$$

NOTE : The diagonal values are ignored.

Total Degree Network Centralization

Go To Measures by Report

Informal Name

None

Citation

Freeman, L.C. (1979). Centrality in Social Networks I: Conceptual Clarification. Social Networks, 1, 215-239.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Fast

Description

A centralization of a square network based on the Total-Degree Centrality of each node.

Example : An organization has a sales team, computers, sales data and a marketing team. How central all of these entities are within the organization is its total degree network centralization value.

Square matrix : DataType=real.

Graph Level : DataType=real.

$$\Re \in [0,1]$$

Let N be a unimodal network with n entities.

let

$$d_i = \text{Total Degree Centrality of entity i}$$

let

$$\overline{d} = \max\{d_i \mid 1 \leq i \leq n\}$$

Then

$$\text{Total Degree Network Centralization} = \left(\sum_{1 \leq i \leq n} \overline{d} - d_i \right) / (n - 2)$$

Transitivity

Go To Measures by Report

Informal Name

None

Citation

NetStat

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

The percentage of link pairs { (i,j), (j,k)} in the network such that (i,k) is also an link in the network.

Example : The percentage of pairs in the employee communication network where employee **A** is linked to **B** and **B** is linked to **C** and **C** is also linked to **A**

Input :

Square matrix : DataType=binary.

Output :

Graph Level : DataType=real

$$\Re \in [0,1]$$

Let G = (V,E) be the graph representation of the square network.

let

$$I = \{(i,j,k) \in V^3 \mid i,j,k \text{ distinct } \}$$

let

$$Potential = \{ (i,j,k) \in I \mid (i,j) \in E, \text{ and } (j,k) \in E \}$$

let

$$Complete = \{ (i,j,k) \in Potential \mid (i,k) \in E \}$$

Then

$$Transitivity = |Complete| / |Potential|$$

Triad Count

Informal Name

None

Citation

NetStat

Minimum and Maximum Values

Minimum : 0

Maximum : Integer (N-1) (N-2)

Computational Time

Medium

Description

The number of triads centered at each node in a square network. A triad is a relationship amongst three nodes such that they constitute a distinct and important relationship which is deemed relevant to network architecture.

Input :

Square matrix : DataType=binary.

Output :

Node Level : DataType=integer.

$$Z \in [0, (n-1)(n-2)]$$ for N dimension n.

Let G=(V,E) represent a square network with n entities.

And let Triad be a matrix of dimension n x n.

$$Triad(i,i) = 0$$

$$Triad(i,j) = card\{ \ k \ | \ k \ != i, \ k \ != j; \ AA(i,j) \wedge AA(i,k) \wedge AA(k,j) \ \}, \ i \neq j$$

Then

$$\text{the Triad count for agent } i = sum(Triad(i,:))$$

Upper Boundedness

Go To Measures by Report

Informal Name

None

Citation

Krackhardt, D. 1994. Graph Theoretical Dimensions of Informal Organizations. In Computational Organization Theory, edited by Carley, Kathleen M. and M.J. Prietula. Hillsdale, NJ: Lawrence Erlbaum Associates, 1994.

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Slow

Description

The degree to which pairs of agents have a common ancestor.

Example : In a given organization, if Bob and Ahmed were both hired by Pedro this measure would compute a value correlating to this shared ancestry or linkage that both have to Pedro.

Input :

594

Square matrix : DataType=binary.

Output :

Graph Level : DataType=real.

$$\Re \in [0,1]$$

Watts-Strogatz Clustering Coefficient

Go To Measures by Report

Informal Name

None

Citation

Watts DJ, Strogatz SH "Collective dynamics of 'small-world' networks" NATURE 393 (6684): 440-442 JUN 4 1998.

Minimum and Maximum Values

Values for both Graph Level and Node Level

Minimum : 0

Maximum : 1

Computational Time

Slow

Description

Measures the degree of clustering in a network by averaging the clustering coefficient of each node, which is defined as the density of the node's ego network.

Example : The average of the proportion of links between each employee and his or her direct neighbors divided by the number of links that could possibly exist between them.

The clustering coefficient gives a sense of the local characteristics of the network--how information spreads by means of employee groups. A higher clustering coefficient supports local information diffusion as well as a decentralized infrastructure because employees are likely to share information and know what is happening in their work group.

Input :

Square matrix : DataType=binary.

Output :

Graph Level : DataType=real.

Node Level : DataType=real.

Graph Level: $\Re \in [0,1]$

Entity Level: $\Re \in [0,1]$

let **G=(V,E)** be the graph representation of a square network.

Define for each node $v \in V$ its Clustering Coefficient cc_v:

let Gv = ego network of entity v

Then Clustering Coefficient for entity:

$$\text{entity } v = cc_v = \text{density}(Gv)$$

Then Clustering Coefficient for graph:

$$\text{graph} = \frac{\sum_{v \in V} cc_v}{|V|}$$

Weak Boundary Spanner

Go To Measures by Report

Informal Name

None

Citation

Cormen, Leiserson, Riverest, Stein, 2001

Minimum and Maximum Values

Minimum : 0

Maximum : 1

Computational Time

Medium

Description

A node which if removed from a network creates a new component.

Input :

Square matrix : DataType=binary and Directed=No.

Output :

Node Level : DataType=binary.

Weak Component Count

Go To Measures by Report

Informal Name

None

Citation

Wasserman, Stanley and Katherine Faust. "Social Network Analysis: Methods and Applications." Cambridge: Cambridge University Press, 1994.

Minimum and Maximum Values

Minimum : 0

Maximum : Integer Value

Computational Time

Medium

Description

The number of weakly connected components in an undirected (symmetric) network.

Input :

Square matrix : DataType=binary and Directed=No.

Output :

Graph Level : DataType=integer.

$$Z \in [0, |V|]$$

Given a square, symmetric network **G=(V,E)**, the Weak Component Count is the number of connected components in G. Such components are called **weak** because the graph G is undirected.

Weak Component Members

Go To Measures by Report

Informal Name

None

Citation

Wasserman, Stanley and Katherine Faust. "Social Network Analysis: Methods and Applications." Cambridge: Cambridge University Press, 1994.

Minimum and Maximum Values

Minimum : 0

Maximum : Any Integer Value

Computational Time

Medium

Description

Computes the component to which each node belongs in the underlying, undirected network.

Example : Given a square, **symmetric network G=(V,E)**, the Weak Component Member for a entity is the weak component to which the entity belongs. The components are numbered from 1 to the number of weak components (Weak Component Count).

> **Input :**
>
> > Square matrix : DataType=binary.
>
> **Output :**
>
> > Node Level : DataType=integer.
>
> $$Z \in [0, |V|]$$
>
> > Given a square, **symmetric network G=(V,E)**, the Weak Component Member for a entity is the weak component to which the entity belongs. The components are numbered from 1 to the number of weak components (Weak Component Count).

🔲 Measures Used by Report

Reports and Measures Used

Report	Measures
All Measures	
Computes a collection of measures and generates output in categories. Each category analyzes a particular aspect of the meta-network structure.	
Belief Propagation	
Estimates belief propagation through	

social networks	
Capabilities	
Analyzes the knowledge, resource, and task capabilities of agents and organizations in the network.	
Change in Key Entities	**Actual Workload**, **Betweenness Centrality**, **Clique Count**, **Cognitive Demand**, **Eigenvector Centrality**, **Exclusivity**, **Exclusivity Complete**, **Potential Boundry Spanner**, **Row Degree Centrality**, **Total Degree Centrality**
Analyzes the changes over time of the agents that score highest in measure values.	
Communication Network Assessment	**Betweenness Centrality**, **Boundry Spanner**, **Eigenvector Centrality**, **Exclusivity**, **Potential Boundry Spanner**, **Total Degree Centrality**
This report assesses a Communications network. As information on speed of information transmittal becomes available it will be updated. This report takes a resource by resource network (preferably one where the nodes are Communications devices and the links are the speed or reliability of the link).	
Communicative Power	**Betweenness Centrality**, **Density**, **Link Count**, **Node Count**, **Total Degree Centrality**
Analyzes one or more semantic networks to classify concepts by measure values.	
Communicators	**In-Degree Centrality**, **Out-Degree Centrality**, **Betweenness Centrality**, **Node Count**, Number of Components, **Clique Count**, **Density**, **Watts Strogatz Clustering Coefficient**
Analyzes the Communication network (agent by agent) and gives high level statistics on its structure.	
Context	
Compares measured values against various stylized forms of networks in an effort to characterize network topology.	
Core Network	**Betweenness Centrality**, **Density**,

Computes the core network and reports general statistics.	**Node Count**, **Total Degree Centrality**
CPOF	
Analyzes the creation and modification of events and tasks and agent collaboration over time.	
Critical Sets	
Finds critical sets of nodes in a network that best reach all other nodes, or whose removal maximally disrupts a network.	
Custom	Whatever measures are selected by the user.
Computes a custom report with user selected measures and output tables, pictures, and figures.	
Drill Down	
GeoSpatial Assessment	
Finds nodes that are co-located based, and counts the number of nodes that appear at location pairs.	
Group Talk	**Row Degree Centrality**
Analyzes the relationship between agents and knowledge.	
Health	**Access Redundancy**, **Assignment Redundacy**, **Average Distance**, **Watts Strogatz Clustering Coefficient**, **Density**, **Total Degree Centralization**
Analyzes health department data to find potential problems.	
Hot Topics (Content Analysis)	
Analyzes semantic network output from AutoMap.	
Immediate Impact	**Average Distance**, **Average Speed**, **Betweenness Centrality**, **Watts Strogatz Clustering Coefficient**, **Cognitive Demand**, **Communication Congruence**, **Complexity**, **Density**, **Diffusion**, **Fragmentation**, **Isolate**
Computes the key actors of the network, and then isolates them individually to determine the effect on measure values.	

	Count, Overall Fragmentation **Node Count**, **Performance as Accuracy**, **Total Degree Centrality**
Influence Net	**Betweenness Centrality**, **In Degree Centrality**
Displays the influence network and gives high level statistics on its structure.	
Key Entity	**Actual Workload**, **Authority Centrality**, **Average Speed**, **Betweenness Centrality**, **Clique Count**, **Cognitive Demand**, **Spatial Eigenvector Centrality**, **Communication Congruence**, **Complexity**, **Density**, **Exclusivity**, **Exclusivity Complete**, **Fragmentation**, **Hub Centrality**, **Eigenvector Centrality**, **Knowledge Congruence**, **Location Relevance**, **Performance as Accuracy**, **Potential Boundry Spanner**, **Row Degree Centrality**, **Shared Situation Awareness**, **Spatial Eigenvector Centrality**, **Total Degree Centrality**
Identifies Key Entities and groups who by virtue of their position in the network are critical to its operation.	
Large Scale	
Computes all network analysis measures.	
Local Patterns	Special
Find link patterns in a network, such as Star, Checkerboard, and Clique.	
Locate Subgroup	
Identifies the subgroups present in the network using various grouping algorithms.	
Management	**Actual Workload**, **Authority Centrality**, **Cognitive Demand**, **Column Breadth**, **Complexity**, **Exclusivity**, **Exclusivity Complete**, **Hub Centrality**, **In Degree Centrality**, **Knowledge Congruence**, **Resource Congruence**, **Shared**
Identifies over- and under-performing individuals and assesses the state of the network as a functioning organization.	

	Situation Awareness, **Simmelian Ties**
Merchant Marine Vessel	
Analyzes the relationships between crew members, owners, vessels and locations of the Merchant Marine Vessel data.	
Missing Links	**Average Distance**, **Average Speed**, **Betweenness Centrality**, **Watts Strogatz Clustering Coefficient**, **Communication Congruence**, **Density**, **Diffusion**, **Fragmentation**, **Isolate Count**, Overall Complexity, Overall Fragmentation, **Performance as Accuracy**, **Total Degree Centrality**
Compute possible missing links in a network.	
Optimizer	Special
Adapt the link structure of a meta-network to maximize or minimize selected measure values.	
Part Of Speech	**Betweenness Centrality**, **Total Degree Centrality**
Describe the distribution of concepts by part of speech in one or more networks.	
Potential Errors	**Cognitive Expertise**, **Cognitive Similarity**
Detects potential errors in agent to agent interactions based on the expected interactions from Cognitive Similarity and expertise.	
QAP/MRQAP Analysis	
Computes QAP and MRQAP Correlation and Regression (Dekker and Y-Permutation methods) on input networks.	
Rules of Engagement	Special

This report analyzes the trails that are found in the Trails Viewer.	
Semantic Network	**Betweenness Centrality**, **Watts Strogatz Clustering Coefficient**, **Density**, **Diameter**, **Node Count**
Analyzes one or more semantic networks, computing the central networks and key concepts and links.	
Simmelian Ties	**Betweenness Centrality**, **Connectedness**, **Density**, **Link Count**, **efficiency**, **Hierarchy**, **Potential Boundary Spanner**, **Total Degree Centrality**, **Upper Boundedness**
Calculates the number of Asymmetric, Sole-Symmetric, and Simmelian Ties in unimodal networks computes standard measures on them.	
Shortest Path	
Computes the shortest path between two entities and general statistics on the paths. The sphere of influence of each entity is also computed.	
Sphere of Influence	**Actual Workload**, **Betweenness Centrality**, **Clique Count**, **Cognitive Demand**, **Eigenvector Centrality**, **Exclusivity**, **Exclusivity Complete**, **Potential Boundry Spanner**, **Shared Situation Awareness**, **Total Degree Centrality**
For each individual, identifies the set of actors, groups, knowledge, resources, etc. that influence and are influenced by that actor.	
Standard Network Analysis	**Authority Centrality**, **Average Distance**, **Betweenness Centrality**, **Betweenness Centralization**, **Clique Count**, **Closeness Centrality**, Closeness Centralization, **Watts Strogatz Clustering Coefficient**, **Column Count**, **Communicative Need**, **Connectedness**, **Density**, **Link Count**, **efficiency**, **Eigenvector Centrality**, **Fragmentation**, **Hierarchy**, **Hub Centrality**, **In Degree Centrality**, **Information Centrality**, **Isolate Count**, **Network Levels**, **Out Degree Centrality**, **Row Count**, **Simmelian Ties**, **Total Degree Centrality**, **Total Degree Centralization**, **Upper Boundedness**, **Weak Component Count**

Calculates the standard network analysis measures (Degree Centrality, Betweenness Centrality, etc).	
Statistical Change Detection	**Betweenness Centrality**, **Closeness Centrality**, **Density**
Analyzes the changes in network-level measures over time using a Statistical Process Monitoring (SPM) control chart. Control limits are determined based on a user defined level of risk.	
Statistical Distribution	No predefined measures, except those with node-level output
Tactical Insight	**Betweenness Centrality**, **Total Degree Centrality**
Analyzes the top agent nodes across time periods, tracking locations and measure values.	
Trails	
Analyzes the trails that an entity class makes through another entity class, for example, how vessels pass through ports.	
Trails Analysis	**Betweenness Centrality**, **Clique Count**, **Eigenvector Centrality**, **Exclusivity**, **Potential Boundry Spanner**, **Row Degree Centrality**, **Total Degree Centrality**
This report analyzes the trails that are found in the Trails Viewer.	
Unique Trails	
Analyzes the path of an entity class through locations over time.	

All Measures

Description :

Computes a collection of measures and generates output in categories. Each category analyzes a particular aspect of the meta-network structure.

What networks are in the meta-network :

What is needed to run this report :

Don't run the time consuming measures option :

Output :

Belief Propagation

Description :

Estimates belief propagation through social networks. This report contains the most common beliefs shared by most people, the beliefs with the highest GINI coefficient, the most strongly held beliefs, the most likely to change beliefs, the most neutral individuals, and the most opinionated individuals.

What networks are in the meta-network :

What is needed to run this report :

Don't run the time consuming measures option :

Output :

Single Meta-Network, multiple Meta-Network comparison.

Capabilities

Description :

Analyzes the knowledge, resource, and task capabilities of agents and organizations in the network. This is used to assess an individuals' or organizations' capability to perform tasks.

What networks are in the meta-network :

What is needed to run this report :

Don't run the time consuming measures option :

Output :

Single Meta-Network, side-by-side comparison, multiple Meta-Network comparison.

Change In Key Entities

Description :

Analyzes the changes over time of the agents that score highest in measure values.

This report provides information on the overall structure of a complex network and identifies, for each type of node, which nodes stand out. A complex network typically is a meta-network with multiple types of entities such as people, expertise, resources and locations. Overall structure is defined using meta-network metrics such as Overall Complexity, Social Density, Social Fragmentation, Communication Congruence, Knowledge Congruence, Performance as Accuracy, and Average Communication Speed. Key entities within each of the component networks are identified in terms of metrics such as Degree Centrality, Cognitive Demand, Specialization, and Redundancy. Separate key-entity pages are produced for each type of entity. For example, if there are people and resources - then key people and key resources are both identified. If the user selects two or more networks, this report will compare the results.

What networks are in the meta-network :

What is needed to run this report :

At least two networks.

Don't run the time consuming measures option :

Output :

Single Meta-Network.

607

Communications Network Assessment

Description :

This report assesses a communications network. As information on speed of information transmittal becomes available it will be updated. This report takes a resource by resource network (preferably one where the nodes are communications devices and the links are the speed of reliability of the link).

What networks are in the meta-network :

What is needed to run this report :

Don't run the time consuming measures option :

Output :

Communicative Power

Description :

Analyzes one or more semantic networks to classify concepts by measure values.

What networks are in the meta-network :

What is needed to run this report :

Don't run the time consuming measures option :

Output :

Single Meta-Network, side-by-side comparison, multiple Meta-Network comparison.

Communicators

Description :

Analyzes the communication network (agent by agent) and gives high level statistics on its structure.

What networks are in the meta-network :

What is needed to run this report :

Don't run the time consuming measures option :

Output :

Single Meta-Network, side-by-side comparison.

Context

Description :

Compares measured values against various stylized forms of networks in an effort to characterize network topology.

What networks are in the meta-network :

What is needed to run this report :

Don't run the time consuming measures option :

Output :

Single Meta-Network.

Core Network

Description :

Computes the core network and reports general statistics.

What networks are in the meta-network :

What is needed to run this report :

Don't run the time consuming measures option :

Output :

Single Meta-Network, side-by-side comparison.

Critical Sets

Description :

Finds critical sets of nodes in a network that best reach all other nodes, or whose removal maximally disrupts a network.

What networks are in the meta-network :

What is needed to run this report :

Don't run the time consuming measures option :

Output :

Single Meta-Network.

Custom

Description :

Computes a custom report with user selected measures and output tables, pictures, and figures.

What networks are in the meta-network :

What is needed to run this report :

Don't run the time consuming measures option :

Output :

A single Meta-Network.

Drill Down

Description :

Computes the drill down from a selected organization and then to a selected agent.

What networks are in the meta-network :

What is needed to run this report :

Don't run the time consuming measures option :

Output :

A single Meta-Network.

Geospatial Assessment

Description :

Finds nodes that are co-located based, and counts the number of nodes that appear at location points.

What networks are in the meta-network :

What is needed to run this report :

Don't run the time consuming measures option :

Output :

A single Meta-Network.

Group Talk

Description :

Analyzes the relationship between agents and knowledge.

What networks are in the meta-network :

What is needed to run this report :

Don't run the time consuming measures option :

Output :

A single Meta-Network.

Hot Topics

Description :

This report analyzes semantic network output from AutoMap and provides standard content analysis information. When you have semantic network data or lists of concepts, networks derived from email content, or texts or documents of any sort.

It outputs the most frequent concepts and the least frequent concepts. Given a semantic network, this report displays statistics on the concepts and links in the map, and the distribution of concept frequencies. This report takes unstructured information into account. Running the hot topics report is similar to performing content analysis [1]. The user can vary the number of highest scoring terms to output.

What networks are in the meta-network :

What is needed to run this report :

Don't run the time consuming measures option :

Output :

Single Meta-Network, multiple Meta-Network comparison.

[1] Carley, K.M., Content Analysis, in The Encyclopedia of Language and Linguistics, R.E. Asher, Editor. 1990, Pergamon Press: Edinburgh. p. 725-730.

Immediate Impact

Description :

Computes the key actors of the network, and then isolates them individually to determine the effect on measure values.

This report allows the user to engage in **what if** analysis about the immediate impact of removing one or more nodes. Comparative statistic techniques are used to assess the network with and without the nodes in question.

The user identifies a network, identifies which node or nodes to remove, and then ORA computes the consequent changes in the network by removing those nodes in terms of overall network structure and key entities. A feature of this report is that the user can save the meta-network with the removed nodes and then assess the modified network with any of the other reports.

The Immediate Impact report is designed to help users understand how the removal of certain nodes, or a certain number of nodes, can effect a network. An **Impact Analysis** allows users to select specific nodes in the network to remove, and then compare various metrics between the original network and the network where these specific nodes are removed. In a **Replication Analysis**, users specify a specific number of random nodes to remove from a network. Users can then comparemetrics of the original network with the average of a series of networks from which the set number of random nodes has been removed.

What networks are in the meta-network :

What is needed to run this report :

Don't run the time consuming measures option :

Output :

Single Meta-Network.

Influence Net

Description :

Displays the influence network and gives high level statistics on its structure.

What networks are in the meta-network :

What is needed to run this report :

Don't run the time consuming measures option :

Output :

Single Meta-Network.

Key Entity

Description :

Identifies key entities and groups who by virtue of their position in the network are critical to its operation.

This report performs a network-level report to find Overall Complexity, Social Density, Social Fragmentation, Communication Congruence, Knowledge Congruence, Performance as Accuracy, and Avg Communication Speed. (Replaces the Event Analysis, Intelligence, and Location Analysis Reports).

What networks are in the meta-network :

What is needed to run this report :

Don't run the time consuming measures option :

Output :

Single Meta-Network, side-by-side comparison, multiple Meta-Network comparison.

Large Scale

Description :

Computes all network analysis measures.

What networks are in the meta-network :

What is needed to run this report :

Don't run the time consuming measures option :

Output :

Single Meta-Network.

Local Patterns

Description :

Find link patterns in a network, such as Star, Checkerboard, and Clique.

What networks are in the meta-network :

What is needed to run this report :

Don't run the time consuming measures option :

Output :

Single Meta-Network.

Locate SubGroups

Description :

Identifies the sibgroups present in the network using various grouping algorithms. Most algorithms only work on a single network at a time

What networks are in the meta-network :

What is needed to run this report :

Don't run the time consuming measures option :

Output :

Management

Description :

Identifies over- and under- performing individuals and assesses the state of the network as a functioning organization.

Like some of the other key-entity reports, the Management Report uses a multi-modal network to consider various aspects of an organization and report both organizational-level metrics as well as node-measures. This report includes identification of both top-ranked actors and knowledge, but also least-ranked actors. Thus, this report provides answers to questions such as:

Which actors are most central to the organization?

What knowledge comprise the organization's core competencies?

Which actors have access to the least knowledge?

Which actors are not well-integrated socially?

How much of the knowledge is unique to only one actor (across all actors)?

What networks are in the meta-network :

What is needed to run this report :

Don't run the time consuming measures option :

Output :

Single Meta-Network, multiple Meta-Network comparison.

Merchant Marine

Description :

Analyzes the relationship between crew members, owners, vessels, and locations of the Merchant Marine Vessel data. It is used with a special data-set developed by CASOS.

What networks are in the meta-network :

What is needed to run this report :

Don't run the time consuming measures option :

Output :

Single Meta-Network.

Missing Links

Description :

Compute possible missing links in a network.

What networks are in the meta-network :

What is needed to run this report :

Don't run the time consuming measures option :

Output :

Single Meta-Network.

Network Distribution

Description :

Analyzes a collection of Meta-Networks and show what has changed in terms of nodes and links when going from A to B. It is summerized in the following table.

Uninvolved/Unattached	nodes and links in A that are not in B	A - B
Common	nodes and links in A and are also in B	B intersect A
New	nodes and links not in A and in B	B - A

What networks are in the meta-network :

What is needed to run this report :

Don't run the time consuming measures option :

Output :

Optimizer

Description :

Adapt the link structure of a Meta-Network to maximize or minimize selected measure values.

What networks are in the meta-network :

What is needed to run this report :

Don't run the time consuming measures option :

Output :

Single Meta-Network.

Part of Speech

Description :

Describe the distribution of concepts by part of speech in one or more networks.

What networks are in the meta-network :

What is needed to run this report :

Don't run the time consuming measures option :

Output :

Single Meta-Network, multiple Meta-Network comparison.

Potential Errors

Description :

Detects potential errors in agent to agent interactions based on the expected interactions from cognitive similarity and expertise.

What networks are in the meta-network :

What is needed to run this report :

Don't run the time consuming measures option :

Output :

Single Meta-Network.

Public Health

Description :

Analyzes health department data to find potential problems.

What networks are in the meta-network :

What is needed to run this report :

Don't run the time consuming measures option :

Output :

Single Meta-Network.

QAP/MRQAP Analysis

Description :

Computes QAP and MRQAP Correlation and Regression (Dekker and Y-Permutation methods) on input networks.

MRQAP : Used for multivariate cases if your dependent variable is continuous or count data (like in a negative binomial case), you should use MRQAP. If your dependent variable is binary, you should use ERGM (P*) which is forthcoming in ORA. One can perform MRQAP on data that have a dichotomous dependent variable (basically, this is equivalent to using a linear probability model). You do need to interpret your results accordingly.

The MRQAP report is setup to compare the networks within a single MetaNetwork: e.g. communication vs department membership. However, you can also select multiple MetaNetworks for MRQAP, but you can not do regression of a network in one MetaNetwork vs a network in a different MetaNetwork.

You can do a regression using networks in the same and or different meta-networks. Moreover, you can take a vector and create a repeated column or repeated row network and use that network in the regression as well.

QAP Analysis : Computes QAP Correlation and Regression (Dekker and Y-Permutation methods) on input matrices. QAP is designed as a bivariate test (only two variables). Generally, QAP is perfectly fine for almost any bivariate network problem.

What networks are in the meta-network :

What is needed to run this report :

Define the independent matrices as well as the dependent network used in the QAP calculations. This is also given a random number seed and number of permutations to run.

Lastly you can set to run any of the following: QAP COrrelation, Dekker Regression, or Y-Permutation Regression analysis.

Don't run the time consuming measures option :

Output :

Single Meta-Network.

Semantic Network

Description :

This report analyzes one or more semantic networks, computing central networks and key concepts and links. It assesses the connection among concepts. Will derive networks from email content, texts, or documents of any sort.

The semantics networks report analyzes one or more semantic networks, computes the central graphs, and key concepts and links. This is a comparison of two or more semantic networks. Each node in

the network is considered a concept, and each link connects two concepts. Link weights are interpreted as the number of times it occurred in the underlying input text.

The report includes tests whether the two semantic networks are statistically different using four different T-tests, based on the number of nodes, the number of links, betweenness centrality, and degree centrality.

The symmetric distance of network A to network B is the number of concepts and links in A that are not in B. In set theoretic terms, this is the set difference (A - B).

This report also provides communicative power analysis. Concepts are classified according to whether they have high and low values for the measures: total degree centrality, betweenness centrality, and consensus. The concept classes are based on [1]:

Ordinary (low degree, low betweenness, low consensus)

Factoids (low degree, low betweenness, high consensus)

Buzzwords (low degree, high betweenness, low consensus)

Emblems (low degree, high betweenness, high consensus)

Allusions (high degree, low betweenness, low consensus)

Stereotypes (high degree, low betweenness, high consensus)

Placeholders (high degree, high betweenness, low consensus)

Symbols (high degree, high betweenness, high consensus)

The report also creates new networks that represent different levels of agreement between the loaded networks. The set-theoretic intersection on the 25, 50, 75% percent level is considered for this comparison. More precisely, the following outputs are computed:

The union of compared links.

The consensus between networks as it is represented by the intersection of links.

The dissension between networks as it is represented as the difference.

The report outputs the nodes and edges in the intersection, offset, and union. Performing map comparison can help to answer questions such as:

Do different people use the same words and themes in the same way?

Do different people link concepts and themes in the same way?

Do different people share the same knowledge?

How similar or different are the analyzed texts?

What networks are in the meta-network :

What is needed to run this report :

Set the percentage of maps a link must be in in order to be included in the central network.

Don't run the time consuming measures option :

Output :

Single Meta-Network, side-by-side comparison.

Select whether to generate hamming distances between Meta-Networks.

[1] Carley, K.M., Network text analysis: The network position of concepts, in Text Analysis for the Social Sciences: Methods for Drawing Statistical Inferences from Texts and Transcripts, C.W. Roberts, Editor. 1997, Lawrence Erlbaum Associates, Inc.: Mahwah, NJ, USA. p. 79-100.

Shortest Path

Description :

Computes the shortest path between two entities and general statistics on the paths. The sphere of influence of each entity is also computed.

Why use this report :

What networks are in the meta-network :

What is needed to run this report :

Don't run the time consuming measures option :

Output :

Single Meta-Network.

Simmelian Ties Analysis

Description :

Calculates the number of Asymmetric, Sole-Symmetric, and Simmelian Ties in unimodal networks computes standard measures on them.

What networks are in the meta-network :

What is needed to run this report :

Single agent by agent network.

Don't run the time consuming measures option :

Output :

Meta-Network with Simmelian Ties.

Sphere of Influence

Description :

For each individual, identifies the set of actors, groups, knowledge, resources, etc. that influence, and are influenced, by that actor.

What networks are in the meta-network :

What is needed to run this report :

Any network. But it's most interpretable if you have an agent by agent and also have another network such as agent by knowledge or agent by resources or agent by location.

Don't run the time consuming measures option :

Output :

Single Meta-Network.

![icon] Standard Network Analysis

Description :

Calculates the standard network analysis measures (Degree Centrality, Betweenness, etc).

This report calculates the standard network analysis measures such as Degree Centrality, Betweeness Centrality, etc. This report can be used on any "square network" where all the nodes are of one type; e.g., a people x people or resource x resource network. The standard network analysis measures typically provide an analyst with a broad overall picture of the structure of the network. If the user selects two or more networks, this report will compare the results.

What networks are in the meta-network :

What is needed to run this report :

Don't run the time consuming measures option :

Output :

Single Meta-Network, side-by-side comparison.

![icon] Statistical Change Detection

Description :

Analyzes the changes in network-level measures over time using a Statistical Process Monitoring (SPM) control chart. Control limits are determined based on a user defined level of risk.

Why use this report :

What networks are in the meta-network :

What is needed to run this report :

Don't run the time consuming measures option :

Output :

Single Meta-Network, multiple Meta-Networks.

Statistical Distribution

Description :

Fits statistical distributions to a network based on measure values.

What networks are in the meta-network :

What is needed to run this report :

Don't run the time consuming measures option :

Output :

Single Meta-Network.

Tactical Insight

Description :

Analyzes the top agent nodes across time periods, tracking locations and measure values.

What networks are in the meta-network :

What is needed to run this report :

>At least three networks.

Don't run the time consuming measures option :

Output :

Single Meta-Network, Multiple Meta-Network comparison.

Trails

Description :

Analyzes the trails that an entity class makes through another entity class, for example, how vessels pass through ports.

What networks are in the meta-network :

What is needed to run this report :

At least two networks. Also requires data in trails format so who/what was where when. Could also be who/what interacted with who/what when.

Don't run the time consuming measures option :

Output :

Single Meta-Network.

Trails Analysis

Description :

Analyzes the trails that are found in the Trails Viewer. It does this by analyzing the trails of one nodeclass as it makes its way through another nodeclass (e.g. how vessels pass through ports).

What networks are in the meta-network :

What is needed to run this report :

Don't run the time consuming measures option :

Output :

Single Meta-Network with multiple Meta-Network comparison.

Unique Trails Report

Description :

Analyzes the path of an entity class through locations over time.

What networks are in the meta-network :

What is needed to run this report :

Don't run the time consuming measures option :

Output :

Single Meta-Network.

	Java Licenses

Library	Name / Website / License
activation	**JavaBeans Activation Framework** **http://java.sun.com/products/archive/javabeans/jaf102.html** **Sun Microsystems, Inc. Binary Code License Agreement**
ant-launcher	**Ant Laucher** **http://ant.apache.org** **The Apache Software License, Version 2.0**
ant-optional-1.5.1	**Ant Optional** **http://ant.apache.org** **The Apache Software License, Version 1.1**
batik-awt-util-1.6	**Batik AWT Utilities** **http://xmlgraphics.apache.org/batik/** **The Apache Software License, Version 2.0**
batik-bridge-1.6	**Batik Bridge** **http://xmlgraphics.apache.org/batik/** **The Apache Software License, Version 2.0**
batik-css-1.6	**Batik CSS** **http://xmlgraphics.apache.org/batik/** **The Apache Software License, Version 2.0**
batik-dom-1.6	**Batik DOM Implementation** **http://xmlgraphics.apache.org/batik/** **The Apache Software License, Version 2.0**
batik-ext-1.6	**Batik Extension** **http://xmlgraphics.apache.org/batik/**

	The Apache Software License, Version 2.0
batik-gvt-1.6	**Batik Graphics Vector Tree** **http://xmlgraphics.apache.org/batik/** **The Apache Software License, Version 2.0**
batik-parser-1.6	**Batik Parser** **http://xmlgraphics.apache.org/batik/** **The Apache Software License, Version 2.0**
batik-script-1.6	**Batik Script** **http://xmlgraphics.apache.org/batik/** **The Apache Software License, Version 2.0**
batik-svg-dom-1.6	**Batik SVG DOM** **http://xmlgraphics.apache.org/batik/** **The Apache Software License, Version 2.0**
batik-svggen-1.6	**Batik SVG Generator** **http://xmlgraphics.apache.org/batik/** **The Apache Software License, Version 2.0**
batik-transcoder-1.6	**Batik Transcoder** **http://xmlgraphics.apache.org/batik/** **The Apache Software License, Version 2.0**
batik-util-1.6	**Batik Utilities** **http://xmlgraphics.apache.org/batik/** **The Apache Software License, Version 2.0**
>batik-xml-1.6	**Batik XML** **http://xmlgraphics.apache.org/batik/** **The Apache Software License, Version 2.0**
casosparser	**CASOS Parser** **http://www.casos.cs.cmu.edu/projects/ora/**
colt	**Colt** **http://acs.lbl.gov/software/colt/** **CERN, 1999**
common-2.1.0	**Commons** **http://commons.apache.org/** **The Apache Software License, Version 2.0**
commons-beanutils-1.4	**Commons BeanUtils** **http://commons.apache.org/** **The Apache Software License, Version 2.0**
commons-cli-1.2	**Commons CLI - Command Line Interface** **http://commons.apache.org/**

	The Apache Software License, Version 2.0
commons-cli-2.0-gt2-pre1	Commons CLI - Command Line Interface http://commons.apache.org/ The Apache Software License, Version 2.0
commons-codec-1.4	Commons Codec http://commons.apache.org/ The Apache Software License, Version 2.0"
commons-collections-3.1	Commons Collections http://commons.apache.org/ The Apache Software License, Version 2.0"
commons-httpclient-3.1	Commons HttpClient http://commons.apache.org/ The Apache Software License, Version 2.0"
commons-io-1.2	Commons I/O Utilities http://commons.apache.org/ The Apache Software License, Version 2.0"
commons-jxpath-1.2	Commons JXPath http://commons.apache.org/ The Apache Software License, Version 2.0"
commons-lang-2.1	Commons Lang http://commons.apache.org/ The Apache Software License, Version 2.0"
commons-logging-1.1.1	Commons Logging http://commons.apache.org/ The Apache Software License, Version 2.0"
commons-logging-api-1.1.1	Commons Logging API http://commons.apache.org/ The Apache Software License, Version 2.0"
commons-math-1.2	Commons Math http://commons.apache.org/ The Apache Software License, Version 2.0"
commons-pool-1.3	Commons Pool http://commons.apache.org/ The Apache Software License, Version 2.0"
concurrent-1.3.4	Java Concurrency http://download.oracle.com/javase/1.5.0/docs/api/java/util/concurrent/package-summary.html Java 2 Platform SE 5.0
db2jcc_dummy	DB2 Spatial Data

-8.2.1	**http://geotools.org/** **Lesser General Public License (LGPL)**
dnsjava-2.0.8	**DNS in Java** **http://www.xbill.org/dnsjava/** **BSD license**
dom4j-1.6.1	
ecore-2.1.0	**Ecore** **http://java2ecore.blogspot.com/** **Lesser General Public License (LGPL)**
facebook-java-api-1.8-final	**Facebook Java API** **http://mrepo.happyfern.com/sites/facebook-java-api/** **MIT License**
facebook-util-1.8-final	**Facebook Companion Utility** **http://mrepo.happyfern.com/sites/facebook-java-api/** **MIT License**
freehep-graphics2d-2.0	**FreeHEP 2D Graphics** **http://java.freehep.org/** **GNU LESSER GENERAL PUBLIC LICENSE, Version 2.1**
freehep-graphicsio-2.0	**FreeHEP GraphicsIO** **http://java.freehep.org/** **GNU LESSER GENERAL PUBLIC LICENSE, Version 2.1**
freehep-graphicsio-pdf-2.0	**FreeHEP GraphicsIO PDF** **http://java.freehep.org/** **GNU LESSER GENERAL PUBLIC LICENSE, Version 2.1**
freehep-graphicsio-svg-2.0	**FreeHEP GraphicsIO SVG** **http://java.freehep.org/** **GNU LESSER GENERAL PUBLIC LICENSE, Version 2.1**
freehep-io-2.0.1	**FreeHEP IO** **http://java.freehep.org/** **GNU LESSER GENERAL PUBLIC LICENSE, Version 2.1**
freehep-util-2.0.1	**FreeHEP Utilities** **http://java.freehep.org/** **GNU LESSER GENERAL PUBLIC LICENSE, Version 2.1**
freehep-xml-2.0.1	**FreeHEP XML** **http://java.freehep.org/** **GNU LESSER GENERAL PUBLIC LICENSE, Version 2.1**
geoapi-nogenerics-2.1-	**GeoAPI** **http://www.geoapi.org/**

M2	GNU Lesser General Public License (LGPL)
geowidgets-1.0-M1	**GeoWidgets** http://geowidgets.sourceforge.net/ **GNU Lesser General Public License (LGPL)**
geronimo-stax-api_1.0_spec-1.0	
gluegen-rt	**GlueGen** http://jogamp.org/gluegen/www/ **BSD license**
gt2-api-2.3.3	**GeoTools API** http://geotools.org/ **Lesser General Public License (LGPL)**
gt2-arcgrid-2.3.3	**GeoTools ArcGrid** http://geotools.org/ **Lesser General Public License (LGPL)**
gt2-arcsde-2.3.3	**GeoTools ArcSDE** http://geotools.org/ **Lesser General Public License (LGPL)**
gt2-brewer-2.3.3	**GeoTools Brewer** http://geotools.org/ **Lesser General Public License (LGPL)**
gt2-coverage-2.3.3	**GeoTools Coverage** http://geotools.org/ **Lesser General Public License (LGPL)**
gt2-coveragetools-2.3.3	**GeoTools Coverage Tools** http://geotools.org/ **Lesser General Public License (LGPL)**
gt2-cql-2.3.3	**GeoTools CQL** http://geotools.org/ **Lesser General Public License (LGPL)**
gt2-db2-2.3.3	**GeoTools DB2** http://geotools.org/ **Lesser General Public License (LGPL)**
gt2-demo-data-2.3.3	**GeoTools Data Demo** http://geotools.org/ **Lesser General Public License (LGPL)**
gt2-demo-mappane-2.3.3	**GeoTools MapPane** http://geotools.org/

	Lesser General Public License (LGPL)
gt2-demo-property-2.3.3	**GeoTools Property Demo** http://geotools.org/ Lesser General Public License (LGPL)
gt2-directory-2.3.3	**GeoTools Directory** http://geotools.org/ Lesser General Public License (LGPL)
gt2-epsg-access-2.3.3	**GeoTools EPSG Access** http://geotools.org/ Lesser General Public License (LGPL)
gt2-epsg-hsql-2.3.3	**GeoTools EPSG-HSQL** http://geotools.org/ Lesser General Public License (LGPL)
gt2-epsg-postgresql-2.3.3	**GeoTools ESPG PostreSQL** http://geotools.org/ Lesser General Public License (LGPL)
gt2-epsg-wkt-2.3.3	**GeoTools ESPF-WKT** http://geotools.org/ Lesser General Public License (LGPL)
gt2-geomedia-2.3.3	**GeoTools Geomedia** http://geotools.org/ Lesser General Public License (LGPL)
gt2-geotiff-2.3.3	**GeoTools GeoTIFF** http://geotools.org/ Lesser General Public License (LGPL)
gt2-gml-2.3.3	**GeoTools GML** http://geotools.org/ Lesser General Public License (LGPL)
gt2-go-2.3.3	**GeoTools Geospatial Object** http://geotools.org/ Lesser General Public License (LGPL)
gt2-graph-2.3.3	**GeoTools Graph** http://geotools.org/ Lesser General Public License (LGPL)
gt2-gtopo30-2.3.3	**GeoTools GTopo30** http://geotools.org/ Lesser General Public License (LGPL)
gt2-hsql-2.3.3	**GeoTools HQSL** http://geotools.org/

	Lesser General Public License (LGPL)
gt2-image-2.3.3	**GeoTools Image** **http://geotools.org/** **Lesser General Public License (LGPL)**
gt2-imagemosaic-2.3.3	**GeoTools ImageMosaic** **http://geotools.org/** **Lesser General Public License (LGPL)**
gt2-imagepyramid-2.3.3	**GeoTools ImagePyramid** **http://geotools.org/** **Lesser General Public License (LGPL)**
gt2-indexed-shapefile-2.3.3	**GeoTools Indexed Shapefile** **http://geotools.org/** **Lesser General Public License (LGPL)**
gt2-main-2.3.3	**GeoTools Main** **http://geotools.org/** **Lesser General Public License (LGPL)**
gt2-mappane-2.3.3	**GeoTools MapPane** **http://geotools.org/** **Lesser General Public License (LGPL)**
gt2-mif-2.3.3	**GeoTools MIFFile** **http://geotools.org/** **Lesser General Public License (LGPL)**
gt2-mysql-2.3.3	**GeoTools MySQL** **http://geotools.org/** **Lesser General Public License (LGPL)**
gt2-openoffice-2.3.3	**GeoTools OpenOffice** **http://geotools.org/** **Lesser General Public License (LGPL)**
gt2-oracle-spatial-2.3.3	**GeoTools Oracle Spatial** **http://geotools.org/** **Lesser General Public License (LGPL)**
gt2-postgis-2.3.3	**GeoTools PostGIS** **http://geotools.org/** **Lesser General Public License (LGPL)**
gt2-referencing-2.3.3	**GeoTools Referencing** **http://geotools.org/** **Lesser General Public License (LGPL)**
gt2-referencing3D-	**GeoTools Referencing 3D** **http://geotools.org/**

2.3.3	**Lesser General Public License (LGPL)**
gt2-render-2.3.3	**GeoTools Render** **http://geotools.org/** **Lesser General Public License (LGPL)**
gt2-sde-dummy-2.3.3	**GeoTools SDE Dummy** **http://geotools.org/** **Lesser General Public License (LGPL)**
gt2-shapefile-2.3.3	**GeoTools Shapefile** **http://geotools.org/** **Lesser General Public License (LGPL)**
gt2-shapefile-renderer-2.3.3	**GeoTools Shapefile Renderer** **http://geotools.org/** **Lesser General Public License (LGPL)**
gt2-svgsupport-2.3.3	**GeoTools SVG Support** **http://geotools.org/** **Lesser General Public License (LGPL)**
gt2-tiger-2.3.3	**GeoTools Tiger** **http://geotools.org/** **Lesser General Public License (LGPL)**
gt2-validation-2.3.3	**GeoTools Validation** **http://geotools.org/** **Lesser General Public License (LGPL)**
gt2-vpf-2.3.3	**GeoTools Vector Product Format** **http://geotools.org/** **Lesser General Public License (LGPL)**
gt2-wfs-2.3.3	**GeoTools Website Feature Server** **http://geotools.org/** **Lesser General Public License (LGPL)**
gt2-widgets-swing-2.3.3	**GeoTools Swing Widgets** **http://geotools.org/** **Lesser General Public License (LGPL)**
gt2-wms-2.3.3	**GeoTools Web Map Service** **http://geotools.org/** **Lesser General Public License (LGPL)**
gt2-xml-2.3.3	**GeoTools XML** **http://geotools.org/>** **Lesser General Public License (LGPL)**
gt2-xml-filter-2.3.3	**GeoTools XML Filter** **http://geotools.org/**

	Lesser General Public License (LGPL)
gt2-xml-gml2-2.3.3	**GeoTools XML GML 2** **http://geotools.org/** **Lesser General Public License (LGPL)**
gt2-xml-gml3-2.3.3	**GeoTools XML GML 3** **http://geotools.org/** **Lesser General Public License (LGPL)**
hsqldb-1.8.0.1	**HyperSQL DataBase** **http://hsqldb.org/** **BSD license**
htmlparser	**HTML Parser** **http://htmlparser.sourceforge.net/** **Common Public License Version 1.0**
httpclient-4.0-beta2	**HttpClient** **http://hc.apache.org/httpcomponents-client-ga/** **Apache License, Version 2.0**
httpcore-4.0-beta3	**HttpCore** **http://hc.apache.org/httpcomponents-core-ga/index.html** **Apache License, Version 2.0**
httpcore-nio-4.0-beta3	**HttpCore Non-Blocking IO** **http://hc.apache.org/httpcomponents-core-ga/httpcore-nio/index.html** **Apache License, Version 2.0"**
httpmime-4.0-beta2	**HttpMime** **http://hc.apache.org/httpcomponents-client-ga/httpmime/index.html** **Apache License, Version 2.0"**
idw	**IntelliDraw Vector Graphic** **Adobe License**
imageioext-asciigrid-1.0-rc1	**Image IO Extentions ASCII Grid** **http://java.net/projects/imageio-ext/** **Lesser General Public License (LGPL)**
imageioext-customstreams-1.0-rc1	**Image IO Extensions Custom Streams** **http://java.net/projects/imageio-ext/** **Lesser General Public License (LGPL)**
iText-2.1.0	**iText** **http://itextpdf.com/** **Affero General Public License**

j3dcore	**Java3D Core** **http://www.j3d.org/** **Java Distribution License**
j3dutils	**Java3D Utilities** **http://www.j3d.org/** **Java Distribution License**
j3d-vrml97	**Java3D VRML** **http://www.j3d.org/** **Java Distribution License**
jai_codec	**Java Advanced Imaging Codec** **http://java.net/projects/jai/** **Sun Microsystems, Inc. Binary Code License Agreement**
jai_core	**Java Advanced Imaging Core** **http://java.net/projects/jai/** **Sun Microsystems, Inc. Binary Code License Agreement**
Jama-1.0.1	**Java Matrix Package** **http://math.nist.gov/javanumerics/jama/** **GNU Lesser General Public License**
java-getopt-1.0.13	**Java GetOpt** **http://javagetopt.sourceforge.net/** **Lesser General Public License (LGPL)**
javaml-0.0.12	**Java Machine Learning Library** **http://java-ml.sourceforge.net/** **GNU Lesser General Public License**
jaxb-api-1.3	**Java Architecture for XML Binding API** **http://jaxb.java.net/** **GNU Lesser General Public License**
jaxb-impl-1.3	**Java Architecture for XML Binding Implementation** **http://jaxb.java.net/** **GNU Lesser General Public License**
jaxb-libs-1.3	**Java Architecture for XML Binding Libraries** **http://jaxb.java.net/** **GNU Lesser General Public License**
JAXWS2.1.3-20071218	**Java API for XML Web Services** **http://jax-ws.java.net/** **GNU Lesser General Public License**
jcalendar	**JCalendar** **http://www.toedter.com/en/jcalendar/index.html** **GNU Lesser General Public License**

jcommon-1.0.9	**JCommon** **http://www.jfree.org/jcommon/** **GNU Lesser General Public License**
jdbc-stdext-2.0	**Java Database Connectivity** **http://www.oracle.com/technetwork/java/** **javase/tech/index-jsp-136101.html** **Java 2 Platform SE 5.0**
jdom-1.1	**JDOM** **http://www.jdom.org/** **JDOM License**
jfreechart-1.0.13	**JFreeChart** **http://www.jfree.org/jfreechart/** **GNU Lesser General Public License**
jfreechart-1.0.13-swt	**JFreeChart SWT** **http://www.jfree.org/jfreechart/** **GNU Lesser General Public License**
jgraphlayout	**Java Graph Layout** **http://www.jgraph.com/jgraph.html** **BSD license**
jhall	**JavaHelp** **http://javahelp.java.net/** **Sun Binary Code License Agreement**
jide-oss-3.0.2	**JIDE Common Layer** **http://www.jidesoft.com/products/oss.htm** **GNU Lesser General Public License**
jlfgr-1.0	**Java Look and Feel Graphics Repository** **http://java.sun.com/developer/techDocs/hi/repository/** **Java 2 Platform SE 5.0**
jogl	**Java OpenGL** **http://java.net/projects/jogl/** **BSD license**
jsi-generics	**Java Spatial Index** **http://jsi.sourceforge.net/** **GNU Lesser General Public License**
json-1.0	**JSON in Java** **http://www.json.org/java/** **The JSON License**
jsr108-0.01	**Java Specification Request** **http://www.jcp.org/en/jsr/detail?id=108**

	Apache License, Version 2.0
jta-1.0.1B	**Java Transaction API** http://java.sun.com/products/jta/ **Java 2 Platform SE 5.0**
jts-1.7-generics	**Java Topology Suite Generics** http://www.vividsolutions.com/jts/JTSHome.htm **Lesser General Public License (LGPL)**
jung	**Java Universal Network/Graph Framework** http://jung.sourceforge.net **The JUNG License**
junit	**JUnit Testing Framework** http://junit.org **JUnit Common Public License - v 1.0**
junit-3.8.1	**JUnit Testing Framework** http://junit.org **JUnit Common Public License - v 1.0**
kxml2-2.3.0	**kXML 2** http://kxml.sourceforge.net/kxml2/ **BSD license**
linkedin-j-android	**LinkedIn for Android Java Wrapper** http://m.linkedin.com/android **The Apache Software License, Version 2.0**
log4j-1.2.6	**Log4j** http://logging.apache.org/log4j/1.2/ **The Apache Software License, Version 2.0**
mail	**JavaMail** http://www.oracle.com/technetwork/java/javamail/index.html **JavaMail API**
MDateSelector14-00347	**MDate Selector** http://web.ukonline.co.uk/mseries **GNU Library or Lesser General Public License (LGPL)**
omcorba	**OpenMap CORBA Classes** http://openmap.bbn.com/ **OpenMap Software License**
omj3d	**Java 3D Classes** http://www.j3d.org/ **Java Distribution License**
omsvg	

	http://openmap.bbn.com/ **OpenMap Software License**
opencsv-1.7	**Open Source CSV Parser** **http://opencsv.sourceforge.net/** **The Apache Software License, Version 2.0"**
openmap-4.6.5	**OpenMap** **http://openmap.bbn.com/** **OpenMap Software License**
picocontainer-1.2	**PicoContainer** **http://picocontainer.org/** **BSD License**
poi-3.7-20101029	**Apache POI** **http://poi.apache.org/** **The Apache Software License, Version 2.0**
poi-ooxml-3.7-20101029	**Apache POI** **http://poi.apache.org/** **The Apache Software License, Version 2.0**
poi-ooxml-schemas-3.7-20101029	**Apache POI** **http://poi.apache.org/** **The Apache Software License, Version 2.0**
poi-scratchpad-3.7.20101029	**Apache POI Scratchpad** **http://poi.apache.org/** **The Apache Software License, Version 2.0"**
postgis-driver-1.0	**PostGIS** **http://postgis.refractions.net/** **GNU General Public License**
postgresql-8.1-407.jdbc3	**PostgreSQL JDBC Driver** **http://jdbc.postgresql.org/** **BSD License**
relaxngDatatype-X	**Oracle Containers for Java** **http://www.oracle.com/technetwork/middleware/ias/downloads/utilsoft-090603.html** **Java 2 Platform SE 5.0**
rome-1.0	**Rome** **http://java.net/projects/rome/** **Apache Software License Version 1.1**
rssutils	**RSS Utilities** **http://java.sun.com/developer/technicalArticles/javaserverpages/rss_utilities/** **Java 2 Platform SE 5.0**

signpost-core-1.2.1.1	**Signpost** **http://code.google.com/p/oauth-signpost/** **The Apache Software License, Version 2.0"**
spatialdb-0.1	
svgSalamander-tiny	**SVG Salamander** **http://svgsalamander.java.net/** **Lesser General Public License (LGPL)**
swingx-1.0	**SwingX 1.0** **http://www.swinglabs.org/** **Lesser General Public License (LGPL)**
TableLayout	**TableLayout** **http://java.sun.com/products/jfc/tsc/articles/tablelayout/** **Java 2 Platform SE 5.0**
toolkits_1-2-0	
trove-2.0.4	**Trove** **http://trove.starlight-systems.com/** **Lesser GNU Public License (LGPL)**
twitter4j-1.1.7	**Twitter4J** **http://twitter4j.org/en/index.html** **The Apache Software License, Version 2.0**
vecmath-1.3.1	**Java 3D Vecmath** **http://java.sun.com/products/java-media/3D/** **Java 2 Platform SE 5.0**
velocity-1.4	**Velocity** **http://velocity.apache.org/,"Th** **Apache Software License, Version 2.0"**
worldwind-0.6.430.13251	**NASA World Wind** **http://worldwind.arc.nasa.gov/java/** **NASA OPEN SOURCE AGREEMENT VERSION 1.3**
velocity-dep-1.4	**Velocity** **http://velocity.apache.org/** **The Apache Software License, Version 2.0"**
xercesImpl-2.7.1	**Xerces Implementation** **http://xerces.apache.org/xerces2-j/** **The Apache Software License, Version 2.0"**

xml-apis-1.0.b2	XML APIs **http://xml.apache.org/commons/** The Apache Software License, Version 2.0"
xml-apis-xerces-2.7.1	Xerces XML APIs **http://xerces.apache.org/xerces2-j/install.html** The Apache Software License, Version 2.0"
xml-beans-2.3.0	XML APIs **http://xml.apache.org/commons/** The Apache Software License, Version 2.0"
xml-writer	XML Writer **http://download.oracle.com/javase/1.5.0/docs/api/javax/sql/rowset/spi/XmlWriter.html** Java 2 Platform SE 5.0
xsd-2.1.1	XML Schema Definition **http://vmgump.apache.org/gump/public/msv/xsdlib/index.html** JAVA WEB SERVICES DEVELOPER PACK, VERSION 1.5
xsdlib-20050614	XML Schema Definition Library **http://vmgump.apache.org/gump/public/msv/xsdlib/index.html** JAVA WEB SERVICES DEVELOPER PACK, VERSION 1.5"

Bibliography

Ashworth, Michael J., & Carley, Kathleen M. (2003). *Critical Human Capital.* Working Paper. Carnegie Mellon University, School of Computer Science, Institute of Software Research, Center for Computational Analysis of Social and Organizational Systems (CASOS).

Bonaich, Philip. (1987). Power and Centrality: A Family of Measures. *American Journal of Sociology, 92*(5), 1170-1182.

Borgatti, Steve P. (2003). The Key Player Problem. In Ronald Breiger, Kathleen M. Carley & P. Pattison (Eds.), *Dynamic Social Network Modeling and Analysis: 2002 Workshop Summary and Papers* (pp. 241-252). Washington, D.C.: National Academies Press.

Burt, Ronald S. (1992). *Structural Holes: The Social Structure of Competition.* Cambridge, MA: Harvard University Press.

Carley, Kathleen M. (2002). *Summary of Key Network Measures for Characterizing Organizational Architectures.* Unpublished document. Carnegie Mellon University, School of Computer Science, Institute for Software Research.

Cataldo, Marcelo, Wagstrom, Patrick, Herbsleb, James, & Carley, Kathleen M. (2006). *Identification of Coordination Requirements: Implications for the design of collaboration and awareness tools.* Paper presented at the Conference on Computer Supported Cooperative Work, Banff, Alberta, Canada. Awarded Best Papers of CSCW 2006.

Cormen, Thomas H., Leiserson, Charles E., Rivest, Ronald L., & Stein, Clifford. (2001). *Introduction to Algorithms.* Cambridge, MA: The MIT Press.

Dekker, David, Kathleen M. Carley, and David Krackhardt. (2002, June 23). *How Do Social Networks Affect Organizational Knowledge Utilization?* Paper presented at the CASOS Conference 2002, Pittsburgh, PA.

Fienberg, Stephen E., Meyer, Michael M., & Wasserman, Stanley S. (1985). Statistical Analysis of Multiple Sociometric Relations. *Journal of the American Statistical Association, 80*, 51-67.

Freeman, Linton C. (1979). Centrality in social networks I: Conceptual clarification. *Social Networks(1)*, 215-239.

Friedkin, Noah E. (1998). *A Structural Theory of Social Influence.* New York: Cambridge University Press.

Krackhardt, David. (1994). Graph Theoretical Dimensions of Informal Organizations. In Kathleen M. Carley & Michael J. Prietula (Eds.), *Computational Organization Theory* (pp. 89-112). Hillsdale, NJ: Lawrence Earlbaum Associates.

Krackhardt, David. (1996). Simmelian Ties: Super Strong and Sticky. In Roderick Kramer and Margaret Neale (Ed.), *Power and Influence in Organizations* (pp. 21-38). Thousand Oaks, CA: Sage.

Latora, Vito, & Marchiori, Massimo. (2001). Efficient behavior of small-world networks. *Physical Review Letters, 87*(19).

Newman, Mark. E. J., Moore, C., & Watts, D. J. (2000). Mean-field solution of the small-world network model. *Physical Review Letters, 84*(14), 3201-3204.

Newman, Mark E. J., & Watts, D. J. (1999). Renormalization group analysis of the small-world network model. *Physics Letters A, 263*(4-6), 341-346.

Newman, M. E. J., & Watts, D.J. (1999). Scaling and percolation in the small-world network model. *Physical Review E, 60*(6), 7332-7342.

Wasserman, Stanley, & Faust, Katherine. (1994). *Social Network Analysis: Methods and Applications.* Cambridge: University of Cambridge Press.

Watts, Duncan J. (1999). Networks, Dynamics, and the Small-World Phenomenon. *The American Journal of Sociology, 105*(2), 493-527.

Watts, Duncan J., & Strogatz, S. H. (1998). Collective dynamics of'small-world' networks. *Nature, 393*(6684), 440-442.

Mixture of Topological Forms

Amblard, Frédeacute;ric. (2002). *Which ties to choose? A survey of social network models for agent-based social simulations.* Paper presented at the International Conference on Artificial Intelligence: Simulation and Planning in High Autonomy Systems, Lisbon, Portugal.

Ayers, W.L. (1938). Some elementary aspects of topology. *The American Mathematical Monthly, 45*(2), 88-92.

Borgatti, Steve P. (1997). Communication Structure and its Effects on Task Performance, from www.analytictech.com/networks/commstruc.htm

Burt, Ronald S. (1980). Models of network structure. *Annual Review of Sociology, 6*, 79-141.

Kephart, J.O. (1994). How topology affects population dynamics. In C. G. Langton (Ed.), *Artificial Life III* (pp. 447-463). New York: Harper & Brothers.

Meyer, John W., & Rowan, Brian. (1977). Institutionalized Organizations: Formal Structure as Myth and Ceremony. *The American Journal of Sociology, 83*, 340-363.

Miles, Raymond E., & Snow, Charles C. (1992). Causes of failure in network organizations. *California Management Review, 34*(4), 53-72.

Moreno, J. L., & Jennings, H. H. (1938). Statistics of Social Configurations. *Sociometry, 1*(3/4), 342-374.

Newman, Mark E. J. (2002). The structure and function of networks. *Computer Physics Communications, 147*, 40-45.

Strogatz, Steven H. (2001). Exploring complex networks. *Nature, 410*, 268-276.

Circle / Wheel

Gilchrist, Jack C., Shaw, Marvin E., & Walker, L. C. (1954). Some effects of unequal distribution of information in a wheel group structure. *Journal of Abnormal Psychology, 49*, 554-556.

Guetzkow, Harold, & Simon, Herbert A. (1955). The impact of certain communication nets upon organization and performance in task-oriented groups. *Organization Science, 1*, 233-250.

Matrix

Burns, Lawton R., & Wholey, Douglas R. (1993). Adoption and abandonment of matrix management programs: Effects of organizational characteristics and interorganizational networks. *Academy of Management Journal, 36*, 106-138.

Kuprenas, John A. (2003). Implementation and performance of a matrix organization structure. *International Journal of Project Management, 21*, 51-62.

Mee, John F. (1964). Matrix Organization. *Business Horizons, 7*, 70-72.

Hierarchy

Miles, Raymond E., & Snow, Charles C. (1995). The new network firm: A spherical structure built on human investment philosophy. *Organizational Dynamics, 23*(4), 5-18.

Ravasz, Erzsebet, & Barabasi, Albert-Laszlo. (2003). Hierarchical organization in complex networks. *Physical Review E, 67*, 026112-1 - 026112-7.

Thorelli, H. B. (1986). Networks: Between markets and hierarchies. *Strategic Management Journal, 7*, 37-51.

Core-Periphery

Borgatti, Stephen P. , & Everett, Martin G. . (1999). Models of core/periphery structures. *Social Networks, 21*, 375-395.

Bourgeois, Michael, & Friedkin, Noah E. (2001). The distant core: Social solidarity, social distance and interpersonal ties in core-periphery structures. *Social Networks, 23*, 245-260.

Random

Newman, Mark E. J. (2003). Random graphs as models of networks. In S. Bornholdt and H. G. Schuster (Ed.), *Handbook of Graphs and Networks* (pp. 35-46). Berlin: Wiley-VCH.

Newman, Mark E. J., Watts, Duncan, & Strogatz, S. H. (2002). Random graph models of social networks *National Academy of Science* (Vol. 99, pp. 2566-2572).

Small-World

Kleinberg, Jon. (2000). The Small-World Phenomenon: An Algorithmic Perspective. *In Annual Proceedings of the Thirty-second Annual ACM Symposium on Theory of Computing* (pp. 163-170). New York: ACM Press.

Milgrim, Stanley. (1967). The Small World Problem. *Psychology Today, 1*, 61-67.

Newman, Mark. (2000). Models of the small world. Journal of *Statistical Physics, 101*, 819-841.

Travers, Jeffrey, and Stanley Milgram. (1969). An experimental study of the small world problem. *Sociometry, 32*, 425-443.

Watts, Duncan J. (1999). Networks, Dynamics, and the Small-World Phenomenon. *The American Journal of Sociology, 105*(2), 493-527.

Watts, Duncan J., & Strogatz, S. H. (1998). Collective Dynamics of "Small-World" Networks. *Nature, 393*, 440-442.

Bow-Tie

Broder, Andrei, Kumar, Ravi, Maghoul, Farzin, Raghavan, Prabhakar, Rajagopalan, Sridhar, Stata, Raymie, . . . Wiener, Janet. (2000). Graph structure in the web. [IBM Research Paper]. *Nature, 405*, 113.

Cellular

Carley, Kathleen M. (2003). Dynamic Network Analysis. In Ronald Breiger, Kathleen Carley & Philippa Pattison (Eds.), *Dynamic Social Network Modeling and Analysis: Workshop Summary and Papers, Committee*

on Human Factors (pp. 133-145). Washington, DC: National Research Council.

Carley, Kathleen M., Matthew Dombroski, Maksim Tsvetovat, Jeffrey Reminga, and Natasha Kamneva. (2003). *Destabilizing Dynamic Covert Networks.* Paper presented at the 8th International Command and Control Research and Technology Symposium, National Defense War College, Washington DC.

Scale-Free

Albert, Reka, & Barabasi, Albert-Laszlo. (2000). Topology of evolving Networks: Local events and universality. *Physical Review Letters, 85*(24), 5234-5237.

Barabasi, Albert-Laszlo, & Bonabeau, Eric. (2003). Scale-Free Networks. *Scientific American, 288*(50-59).

Boss, Michael, Elsinger, Helmut, Summer, Martin, & Thurner, Stefan. (2003). *The Network Topology of the Interbank Market.* Working paper. Santa Fe Institute. Santa Fe.

Cohen, Reuven, Havlin, Shlomo, & Ben-Avraham, Daniel. (2002). Structural Properties of Scale-Free Networks. In S. Bornholdt & H. G. Schuster (Eds.), *Handbook of graphs and networks* (pp. 85-110). Wernheim, Germany: Wiley-VCH.

Dorogovtsev, S. N., Mendes, J. F. F., & A. N. Samukhin, A. N. (2000). Structure of Growing Networks with Preferential Linking. *Physical Review Letters, 85*, 4633-4636.

Simon, Herbert A. (1955). On a Class of Skew Distribution Functions. *Biometrika, 42*(3/4), 425-440.

Complex Networks (Statistical Mechanics)

Albert, Reka, & Barabasi, Albert-Laszlo. (2002). Statistical mechanics of complex networks. *Reviews of Modern Physics, 74*, 47-97.

Dorogovtsev, Sergey, & Mendes, J. F. F. (2002). Evolution of networks. *Advances in Physics, 51*, 1079-1187.

Newman, Mark E. J. (2003). The Structure and Function of Complex Networks. *SIAM Review, 45*(2), 167-256.

Pastor-Satorras, Romualdo; Rubi, Miguel; Diaz-Guilera, Albert (Ed.). (2003). *Statistical mechanics of complex networks.* Berlin: Springer.

Network Processes

Jeong, H, Neda, Z, & Barabasi, A. L. (2003). Measuring preferential attachment in evolving networks. *Europhysics Letters, 61*(4), 567-572.

Motter, Adilson E., & Lai, Yin-Cheng. (2002). Cascade-based attacks on complex networks. *Physical Review E, 66*, 065102-1 - 065102-4.

Newman, Mark. J., Jensen, I., & Ziff, R. M. (2001). *Percolation and Epidemics in a two-dimensional small world.* Working Paper 01-09-046. Santa Fe Institute. Santa Fe, NM.

Pastor-Satorras, Romualdo, & Vespignani, Alessandro. (2002). Immunization of Complex Networks. *Physical Review E, 65*(036104), 1-4.

Vragovic, I., E. Louis, and A. Diaz-Guilera. (2005). Efficiency of informational transfer in regular and complex networks. *Physical Review E, 71*, 036122-1 - 036122-9.